

MicroStation to Unity to VR Workflow

Author:

- Name: Andrew Novinc
- Email: Andrew@Novaclade.com
- YouTube: <https://www.youtube.com/watch?v=-q965SGPzzU&list=PL0JBlaeAgD15fFWHx4d6bFTVR6Je9zCpl>
- Date: April 2018

Introduction:

- The “Unity” real time game engine has an impressive world wide following and has built up an extensive asset store that provides a vast amount of user generated content. With the availability of virtual reality headsets such as the HTC Vive, Oculus Rift, and Windows Mixed Reality, Unity has become a compelling tool of choice for architects interested in creating real time interactive virtual environments.
- Unity uses FBX as its native file format which means that MicroStation’s FBX exporter is very well suited to providing Unity the content that it needs to build virtual worlds.
- After some initial setup, the workflow between MicroStation & Unity becomes seamless which means that iterative changes are automatically incorporated into the Unity project with very little additional user input.

Prerequisites:

- HTC Vive Virtual Reality headset: <https://www.htcvive.com/>
- **Unity** Game Engine (Free Personal Edition) (**Version 2018.1**): <https://unity3d.com/get-unity>
- Unity Assets:
 - Asset store: <https://www.assetstore.unity3d.com/en/>
 - **SteamVR** Plugin (**Version 1.2.3**): <https://www.assetstore.unity3d.com/en/#!/content/32647>
 - **VRTK** (Virtual Reality Toolkit) (**Version 3.3.0**): <https://github.com/thestonefox/VRTK>
 - VRTK YouTube Tutorials: <https://www.youtube.com/channel/UCWRk-LEMUNoZxUmY1wO7DBQ/videos>

Prepare the VRTK Project Files:

- Download the VRTK ZIP file from GitHub.
- Unzip the file and place the extracted folder alongside your other Unity Projects.
- Open the VRTK project with your latest version of Unity to upgrade all the files and make them ready for use.

Surface Normal Direction:

- For performance reasons, Unity's standard shader only does single sided rendering which means that the back face of a surface will be rendered invisible.
- MicroStation's Solids are OK but you will need to be careful and make sure that all Surface geometry has its surface normals pointing in the outward direction.
- It is possible within Unity to use a custom shader that turns OFF “Back Face Culling” but this is something that should be used sparingly and only for things such as chain-link fences or tree foliage. This particular feature isn’t covered in this tutorial.

Create Collision Mesh Geometry:

- The only additional MicroStation geometry that you will need to create is a simplified collision mesh that represents the boundary of where the user is allowed to walk or teleport.
- Even though you might be able to make your entire building model a collision mesh, you will get better performance if you use simplified geometry instead.
- Additionally, if you intend to create grabbable objects, you will need to create a separate convex collision mesh for each object that you intend to interact with. You can combine multiple convex collision meshes to make a suitable concave collision mesh.

Export DGN Geometry as FBX Files:

- Setup a series of empty DGN files that Reference your geometry in a number of small logically organised components that are ready for export. If need be, move all Reference files so that the same point of interest is at the origin (X, Y, Z = 0, 0, 0).
- Open each DGN export file and goto [File > Export > FBX] (or keyin: “mdl keyin fblexport dialog fbx”).
- Set “Stroke Tolerance” to about “0.002” (if you have curved objects).
- Turn ON “Invert Y and Z”.
- Export all of your FBX files one at a time.

Create a Unity Project:

- Start Unity and create a new 3D Project.
- If you are going to be accessing the Unity Project from multiple computers on your network:
 - Make sure that the Project is located on a shared network drive.
 - Make sure that the Global Illumination Cache is located on a shared network drive:
 - Goto [Toolbar > Edit > Preferences... > GI Cache]
 - Set [Maximum Cache Size] to [10 GB or greater].
 - Set [Custom Cache Location] to [ON].
 - Set [Cache Folder Location] to [A shared folder on your network drive].
 - Set [Cache Compression] to [ON].
- In the top right corner set Layout to Default.
- In the top right corner of the Scene View, click on the “Gizmos” drop down menu item:
 - Turn off or reduce the size of the “3D Icons”.
(Do this to minimise the size of the VRTK icons in the Scene view. You can delete them all together if desired).
 - Turn off “Show Grid”.
- In the bottom right corner of the Assets content window, drag the slider to the left to enable “List” view.
- Goto [Toolbar > Window > Lighting > Settings] to make sure the Lighting tab is docked next to the Inspector tab.
- Click on the Inspector tab to make it active.
- Goto [Toolbar > Window > Asset Store] to make sure the Asset Store tab is docked next to the Game tab.
- From the Asset Store, install “**SteamVR Plugin**” and accept all defaults.
- Close the Asset Store tab or click on the Scene tab to make it active.
- Import “**VRTK**” (Virtual Reality Toolkit):
 - Open a second instance of Unity.
 - Open the “**VRTK-master**” project.
 - From the Assets folder, drag and drop the “**VRTK**” folder onto the “**Assets**” folder of your newly created Unity project and then close the second instance of Unity.
 - To entirely remove the VRTK icons from the scene view:
 - Delete the [Projects > VRTK > Source > Editor > Icons] folder.
- Goto [Toolbar > Edit > Project Settings > Player] and in the Inspector tab, set:
 - [Other Settings > Rendering > Color Space] to [Linear].
 - [Other Settings > Rendering > Graphics Jobs (Experimental)] to [ON].
 - [Other Settings > Optimization > Vertex Compression > Tex Coord 0] to [OFF].
 - [Other Settings > Optimization > Optimize Mesh Data] to [OFF].
 - [XR Settings > Virtual Reality Supported] to [ON].
 - [XR Settings > Virtual Reality SDKs] to [OpenVR].
 - [XR Settings > Stereo Rendering Method] to [Single Pass Instanced].
- Goto [Toolbar > Edit > Project Settings > Quality] and in the Inspector tab, set:
 - If your scene has many Realtime light sources you might need to increase:
 - [Rendering > Pixel Light Count] to about [20].
 - [Rendering > Anti Aliasing] to [4x Multi Sampling].
- Goto [Toolbar > File > Save Scene As...]:
 - In the Assets folder, create a new sub-folder called “Scenes”.
 - Double click on the Scenes folder and name the new scene “Main”.

Add Prefabs and Scripts to the Unity Project:

- Make sure that the [Main] scene tree structure is expanded by clicking on the little triangle in the Hierarchy tab.
- Delete the default [Main Camera].
- Setup the VRTK_SDK Manager:
 - In the Hierarchy tab, right click and choose [Create Empty] GameObject. Name it “**VRTK_Files**”.
 - In the Inspector tab, set Position to [0, 0, 0].
 - Click on the [Inspector tab > Add Component] button and search for “**VRTK_SDK Manager**”.
 - In the Hierarchy tab, right click on [VRTK_Files] and choose [Create Empty] GameObject. Name it “**SDK_Files**”.
 - In the Inspector tab, set Position to [0, 0, 0].
 - Drag and drop [Assets > SteamVR > Prefabs > **CameraRig**] onto [VRTK_Files > SDK_Files].
 - Set [CameraRig > Inspector tab > Transform > Position X, Y, Z] to [0, 0, 0].
 - Set [CameraRig > Inspector tab > Steam VR_Play Area > Draw in Game] to [OFF].
 - Set [CameraRig > Inspector tab > Steam VR_Play Area > Size] to [Calibrated].
 - Set [CameraRig > Camera (head) > Inspector tab > Camera > Allow HDR] to [ON].
 - In the Hierarchy tab, click on [SDK_Files] and in the Inspector tab:
 - Click on the [Add Component] button and search for “**VRTK_SDK Setup**”.
 - Set [VRTK_SDK Setup > SDK Selection > Quick Select] to [SteamVR (Standalone: OpenVR)]
 - In the Hierarchy tab, click on [VRTK_Files] and in the Inspector tab:
 - Click on the [VRTK_SDK Manager > Setups > Auto Populat] button.
- Setup VRTK controller scripts:
 - In the Hierarchy tab, right click on [VRTK_Files] and choose [Create Empty] GameObject. Name it “**Script_Files**”.
 - In the Inspector tab, set Position to [0, 0, 0].
 - In the Hierarchy tab, right click on [Script_Files] and choose [Create Empty] GameObject. Name it “**Controller_L**”.
 - In the Inspector tab, set Position to [0, 0, 0].
 - In the Hierarchy tab, right click on [Script_Files] and choose [Create Empty] GameObject. Name it “**Controller_R**”.
 - In the Inspector tab, set Position to [0, 0, 0].
 - In the Hierarchy tab, click on [VRTK_Files]:
 - Drag and drop [Hierarchy tab > VRTK_Files > Script_Files > Controller_L] onto [Inspector tab > VRTK_SDK Manager > Script Aliases > Left Controller].
 - Drag and drop [Hierarchy tab > VRTK_Files > Script_Files > Controller_R] onto [Inspector tab > VRTK_SDK Manager > Script Aliases > Right Controller].
 - Drag and drop [Assets > VRTK > Source > Scripts > Interactions > Interactors > **VRTK_ControllerEvents**] onto [Hierarchy tab > VRTK_Files > Script_Files > Controller_L AND Controller_R].
 - Drag and drop [Assets > VRTK > Source > Scripts > Pointers > **VRTK_Pointer**] onto [Hierarchy tab > VRTK_Files > Script_Files > Controller_L AND Controller_R]:
 - For both controllers, in the Inspector tab, set:
 - [VRTK_Pointer > Pointer Activation Settings > Activation Button] to [Touchpad Press].
 - Drag and drop [Assets > VRTK > Source > Scripts > Pointers > PointerRenderers > **VRTK_BezierPointerRenderer**] onto [Hierarchy tab > VRTK_Files > Script_Files > Controller_L AND Controller_R].
 - Individually for Controller_L and then for Controller_R, in the Inspector tab, drag and drop [VRTK_Bezier Pointer Renderer (Script)] onto [VRTK_Pointer > Pointer Activation Settings > Pointer Renderer].
 - For both controllers, in the [Inspector tab > VRTK_Bezier Pointer Renderer], set:
 - [General Appearance Settings > Valid Collision Color] to about [R:30, G:180, B:220, A:128]
 - [General Appearance Settings > Invalid Collision Color] to about [R:220, G:40, B:40, A:128]
 - [Bezier Pointer Appearance Settings > Tracer Density] to about [25].
 - [Bezier Pointer Renderer Settings > Rescale Tracer] to [ON].
 - [Bezier Pointer Renderer Settings > Cursor Match Target Rotation] to [ON].
 - Drag and drop [Assets > VRTK > Examples > ExampleResources > SharedResources > Prefabs > AnimatedBezierPointer >
 - BezierPointerAnimatedTracer onto [Custom Tracer].
 - BezierPointerCursorBeam onto [Custom Cursor]. (Optional - You could leave this blank)
 - BezierPointerValidCursor onto [Valid Location Object].
 - BezierPointerInvalidCursor onto [Invalid Location Object].
- Setup VRTK teleport scripts:
 - In the Hierarchy tab, right click on [Script_Files] and choose [Create Empty] GameObject. Name it “**PlayArea**”.
 - In the Inspector tab, set Position to [0, 0, 0].
 - Drag and drop [Assets > VRTK > Source > Scripts > Locomotion > **VRTK_DashTeleport**] onto [Hierarchy tab > VRTK_Files > Script_Files > PlayArea] and in the Inspector tab, set:
 - [VRTK_Dash Teleport > Base Settings > Blink Transition Speed] to [0].
 - Note: You can use HeightAdjustTeleport instead of DashTeleport.

Import FBX Files and Set Default Attributes:

- Close Unity and with Windows File Explorer navigate to the [Assets] folder in your Unity project.
 - Create a new sub-folder called "**Geometry**" and copy the previously exported FBX files to this folder.
 - If applicable, Create a new sub-folder called "**Textures**" and copy texture files to this folder that aren't included in the exported FBX files. This might include such things as Normal maps.
- Restart Unity and open your project. Unity will import all new or changed files automatically.
- Goto [Assets > Geometry]. Highlight the FBX collision mesh files and in the Inspector tab, set:
 - [Model > Meshes > Read/Write Enabled] = [OFF]
 - [Model > Meshes > Generate Colliders] = [ON]
 - [Model > Normals & Tangents > Normals] = [None]
 - [Model > Normals & Tangents > Tangents] = [None]
 - [Materials > Import Materials] = [OFF]
 - Press the [**Apply**] button.
- Goto [Assets > Geometry]. Highlight all FBX files except for the collision meshes and in the Inspector tab, set:
 - [Model > Meshes > Read/Write Enabled] = [OFF]
 - [Model > Meshes > Generate Colliders] = [OFF]
 - [Model > Meshes > Generate Lightmap UVs] = [ON] (Used for Lightmap Baking, otherwise turn it OFF)
 - [Advanced > Pack Margin] = [4] (Depending on map resolution, this may need to go up to 20)
 - [Model > Normals & Tangents > Normals] = [Import]
 - [Model > Normals & Tangents > Tangents] = [None] (For better performance)
 - [Materials > Import Materials] = [ON]
 - [Materials > Location] = [Use External Materials (Legacy)]
 - [Materials > Naming] = [From Model's Material].
 - Press the [**Apply**] button.
- Goto [Assets > Geometry > Materials] and tweak the individual material properties.
Note: These tweaks are only done once as all changes are kept even after you replace the FBX files with newer versions.

Add FBX Files to your Scene:

- In the Hierarchy tab, right click and choose [Create Empty] GameObject. Name it "**Model_Files**".
 - In the Inspector tab, set Position to [0, 0, 0].
- From [Assets > Geometry] drag and drop all FBX files onto the "Model_Files" folder created earlier:
 - Make sure the Position Transform for the individual FBX files is set to X, Y, Z = [0, 0, 0].
- Click on the collision mesh in the Hierarchy tab and in the Inspector tab, goto:
 - [Mesh Renderer > Settings cog Icon on the right] and select [Remove Component].
- The [Hierarchy tab > VRTK_Files > SDK_Files > CameraRig] position has already been set to X, Y, Z = [0, 0, 0].
but at this point you can move it to suit the geometry in your scene.

Lighting Setup:

- Goto [Toolbar > Window > Lighting > Settings] and dock the Lighting tab next to the Inspector tab.
- Drag and drop [Hierarchy tab > Main > Directional Light] onto [Lighting tab > Environment > Sun Source].
- At the bottom of the Lighting tab, set [Auto Generate] to [OFF].
- In the Scene Window, move and rotate the “Directional Light” so that it correctly illuminates your project geometry.
- Goto [Toolbar > Edit > Project Settings > Quality] and in the Inspector tab, set:
 - [Shadows > Shadows] to [Hard and Soft Shadows].
 - [Shadows > Shadow Resolution] to [Very High Resolution].
 - [Shadows > Shadow Distance] to [A meaningful value for the scale of your scene].
 - [Shadows > Shadow Cascades] to [Four Cascades].

Option 1 – Realtime Lighting (For Dynamic or Static Objects):

- Click on the “Directional Light” in the Hierarchy tab and in the Inspector tab, set:
 - [Light > Color >] to [Off white. Warm for sun, cool for fill.].
 - [Light > Mode >] to [Realtime].
 - [Light > Shadow Type >] to [Soft Shadows].
 - [Light > Shadow Type > Realtime Shadows > Resolution] to [Very High Resolution].
- In the Lighting tab, set:
 - [Realtime Lighting > Realtime Global Illumination] to [ON].
 - [Mixed Lighting > Baked Global Illumination] to [OFF].
- Press the [Generate Lighting] button at the bottom of the Lighting tab.

Option 2 – Mixed Lighting (Pre-Rendered Baked Lighting for Static Objects):

- Click on the “Directional Light” in the Hierarchy tab and in the Inspector tab, set:
 - [GameObject] to [Static] (Top right corner of Inspector tab)
 - [Light > Color >] to [Off white. Warm for sun, cool for fill].
 - [Light > Mode >] to [Mixed] (or ‘Baked’ if you only have static objects).
 - [Light > Shadow Type >] to [Soft Shadows].
 - [Light > Shadow Type > Realtime Shadows > Resolution] to [Very High Resolution].
- In the Hierarchy tab, highlight all of your geometry that does not move and in the Inspector tab, set:
 - [GameObject] to [Static] (Top right corner of Inspector tab)
- In the Hierarchy tab, one at a time highlight your Static geometry mesh elements and in the Inspector tab, set:
 - [Mesh Renderer > Lighting > Lightmap Settings > Stitch Seams] to [ON].
- In the Lighting tab, set:
 - [Realtime Lighting > Realtime Global Illumination] to [OFF].
 - [Mixed Lighting > Baked Global Illumination] to [ON].
 - In Lightmapping Settings, set:
 - [LightMapper] to [Progressive].
 - [LightMapper > Prioritize View] to [ON].
 - [LightMapper > Direct Samples] to about [64].
 - [LightMapper > Indirect Samples] to about [512].
 - [LightMapper > Bounces] to about [2 – 4].
 - [Lightmap Resolution] to about [64].
 - [Lightmap Padding] to about [4 - 16].
 - [Lightmap Size] to [4096].
 - [Compress Lightmaps] to [OFF].
 - [Ambient Occlusion] to [ON].
 - [Ambient Occlusion > Max Distance] to about [1].
 - [Directional Mode] to [Non-Directional].
 - [Lightmap Parameters] to [Default-HighResolution].
- Press the [Generate Lighting] button at the bottom of the Lighting tab.

Reflection Probe Setup:

- In the Hierarchy tab, right click and choose [Create Empty] GameObject. Name it “**Reflection_Probes**”.
 - In the Inspector tab, set Position to [0, 0, 0].
- Goto [Toolbar > GameObject > Light > Reflection Probe] and place as many Reflection Probes as need be.
- Move the Reflection Probes to their desired positions in the scene and in the Inspector tab, set:
 - [Type] to [Baked].
 - [Runtime Settings > Box Size] to [A size that will encompass the desired geometry in your scene]
 - [Cubemap Capture Settings > Resolution] to [128 or 256]
- Press the [Generate Lighting] button at the bottom of the Lighting tab.

Make Objects Grabbable:

- Drag and drop [Assets > VRTK > Source > Scripts > Interactions > Interactors > **VRTK_InteractTouch**] onto [Hierarchy tab > VRTK_Files > Script_Files > Controller_L AND Controller_R].
- Drag and drop [Assets > VRTK > Source > Scripts > Interactions > Interactors > **VRTK_InteractGrab**] onto [Hierarchy tab > VRTK_Files > Script_Files > Controller_L AND Controller_R]:
 - For both controllers, set:
 - [Inspector tab > VRTK_Interact Grab > Grab Settings > Grab Button] to [Grip Press].
- Highlight the grabbable object in the Hierarchy tab and in the Inspector tab:
 - Click on [Add Component] > [**Rigidbody**].

Option 1 – Simple Box Collider:

- Click on [Add Component] > [**Box Collider**]:
 - Click on the “Edit Collider” button and adjust the size and position of box to suit the geometry.

Option 2 – Convex Mesh Collider:

- Create a convex mesh collider for the grabbable object and import the FBX file just like the main collision mesh.
- Click on [Add Component] > [**Mesh Collider**].
- Set [Mesh Collider > Convex] to [ON].
- Drag and drop [Assets > Geometry > “Grabbable Object Collision Mesh” > Mesh] onto [Inspector tab > Mesh Collider > Mesh].

Option 3 – Concave Mesh Collider:

- There are no “Concave Mesh Colliders” but you can build one up by combining multiple “Convex Mesh Colliders”.
- Create multiple independent convex mesh colliders for the grabbable object and import the FBX files just like the main collision mesh.
- Click on [Add Component] > [**Mesh Collider**]. (Add as many as there are Collision Mesh FBX files)
- Set [Mesh Collider > Convex] to [ON].
- Drag and drop each [Assets > Geometry > “Grabbable Object Collision Mesh” > Mesh] onto its respective [Inspector tab > Mesh Collider > Mesh].
- Drag and drop [Assets > VRTK > Source > Scripts > Interactions > Interactables > **VRTK_InteractiveObject**] onto the grabbable object in the Hierarchy tab and in the Inspector tab, set:
 - [Grab Settings > Is Grabbable] to [ON]
 - [Grab Settings > Hold Button To Grab] to [ON] (or OFF if you prefer)
- Drag and drop [Assets > VRTK > Source > Scripts > Interactions > Interactables > **VRTK_InteractObjectAppearance**] onto the grabbable object in the Hierarchy tab and in the Inspector tab, set:
 - [Grab Appearance Settings > Renderer Visible On Grab] to [OFF]
- Drag and drop [Assets > VRTK > Source > Scripts > Interactions > Interactables > **VRTK_InteractHaptics**] onto the grabbable object in the Hierarchy tab and in the Inspector tab, set:
 - [Haptics On Touch Settings > Strength On Touch] to [0.3]
 - [Haptics On Touch Settings > Duration On Touch] to [0.1]
- Drag and drop [Assets > VRTK > Source > Scripts > Interactions > Interactables > GrabAttachMechanics > **VRTK_ChildOfControllerGrabAttach**] onto the grabbable object in the Hierarchy tab and in the Inspector tab, set:
 - [Base Settings > Precision Grab] to [ON]
- Drag and drop [Assets > VRTK > Source > Scripts > Interactions > Interactables > SecondaryControllerGrabActions > **VRTK_SwapControllerGrabAction**] onto the grabbable object in the Hierarchy tab.

Add Ambient Sound to a Scene:

- Right click on [Assets] and create a new sub folder called “**Audio**”.
- Right click on the “Audio” folder and choose [Import New Asset...].
- Navigate to your music file (usually MP3) and import it.
- In the Hierarchy tab, right click and choose [Create Empty] GameObject. Name it “**Audio**”.
 - In the Inspector tab, set [x, y, z] Position to an appropriate place in your scene.
- In the Inspector tab click on [Add Component] and search for “Audio Source”.
 - Drag and drop [Assets > Audio > Your Audio File] onto [Inspector > Audio Source > AudioClip]
 - Set [Audio Source > Play On Awake] to [ON]
 - Set [Audio Source > Loop] to [ON]
 - Set [Audio Source > Volume] to about [0.2] ... (or whatever suits)
 - Set [Audio Source > Spacial Blend] to [1.0] ... (0.0 for ambient – 1.0 for full 3D spacial)

Create a Floating Menu:

- In this example we will use the Menu to:
 - **[Button]** - Turn ON/OFF different models.
 - **[Button]** - Reset grabbable model position (This might be because you've thrown it out of reach).
 - **[Toggle]** - Play/Stop Audio.
 - **[Button]** - Exit the Scene.
- We are going to need to write three C# Scripts:
 - Right click on [Assets] and choose [Create > Folder]. Name it "**Scripts**".
 - Right click on [Assets > Scripts] and choose [Create > C# Script]. Name it "**MenuToggle**".
 - Double click on the [MenuToggle] script to open it with a code editor (Probably Visual Studio).
 - Delete the default code, paste in the following code, save and close the C# Script file:

```
using UnityEngine;
using VRTK;

public class MenuToggle : MonoBehaviour {
    public VRTK_ControllerEvents controllerEvents;
    public GameObject menu;

    bool menuState = false;

    void OnEnable()
    {
        controllerEvents.ButtonTwoReleased += ControllerEvents_ButtonTwoReleased;
    }

    void OnDisable()
    {
        controllerEvents.ButtonTwoReleased -= ControllerEvents_ButtonTwoReleased;
    }

    private void ControllerEvents_ButtonTwoReleased(object sender, ControllerInteractionEventArgs e)
    {
        menuState = !menuState;
        menu.SetActive(menuState);
    }
}
```

- Right click on [Projects > Assets > Scripts] and choose [Create > C# Script]. Name it "**ResetObjectPosition**".
- Double click on the [ResetObjectPosition] script to open it with a code editor (Probably Visual Studio).
- Delete the default code, paste in the following code, save and close the C# Script file:

```
using UnityEngine;

public class ResetObjectPosition : MonoBehaviour {

    private Vector3 originalPosition;
    private Quaternion originalRotation;

    void Awake()
    {
        this.originalPosition = this.transform.position;
        this.originalRotation = this.transform.rotation;
    }

    public void resetTransform()
    {
        this.transform.position = this.originalPosition;
        this.transform.rotation = this.originalRotation;
    }
}
```

- Right click on [Projects > Assets > Scripts] and choose [Create > C# Script]. Name it "**ExitGame**".
- Double click on the [ExitGame] script to open it with a code editor (Probably Visual Studio).
- Delete the default code, paste in the following code, save and close the C# Script file:

```
using UnityEngine;

public class ExitGame : MonoBehaviour {

    public void doExitGame()
    {
        Application.Quit();
    }
}
```

- In the Hierarchy tab, right click and choose [Create Empty] GameObject. Name it “**Menu_Canvas**”.
 - In the Inspector tab, set Position to [0, 0, 0].
- In the Hierarchy tab, right click on [Menu_Canvas] and choose [Create Empty] GameObject. Name it “**Headset_Follower**”.
 - In the Inspector tab, set Position to [0, 0, 0].
 - Click on the [Add Component] button and search for “**VRTK_SDKObjectAlias**”.
 - Set [SDK Object] to [Headset].
- In the Hierarchy tab, right click on [Menu_Canvas] and choose [Create Empty] GameObject. Name it “**Menu_Follower**”.
 - In the Inspector tab, set Position to [0, 0, 0].
 - Click on the [Add Component] button and search for “**VRTK_TransformFollow**”.
 - Drag and drop [Hierarchy tab > Menu_Canvas > Headset_Follower] onto [Game Object To Follow].
 - Drag and drop [Hierarchy tab > Menu_Canvas > Menu_Follower] onto [Game Object To Change].
 - Set [Position Settings > Smooths Position] to [ON].
 - Set [Position Settings > Max Allowed Per Frame Distance Difference] to about [15].
 - Set [Rotation Settings > Smooths Rotation] to [ON].
 - Set [Rotation Settings > Max Allowed Per Frame Angle Difference] to about [1.5].
- In the Hierarchy tab, right click on [Menu_Follower] and choose [UI > Canvas]. Name it “**Menu**”. In the Inspector tab, set:
 - [Rect Transform > Pos X, Y, Z] to about [0, -0.3, 1.5].
 - [Rect Transform > Width, Height] to about [1024, 1024].
 - [Rect Transform > Scale X, Y, Z] to about [0.001, 0.001, 0.001].
 - [Canvas > Render Mode] to [World Space].
 - Click on the [Add Component] button and search for “**VRTK_UICanvas**”.
- In the Hierarchy tab, right click on [Menu] and choose [UI > Button]. Name it “**Button_1**” and in the Inspector tab, Set:
 - [Pos X, Pos Y, Pos Z] to about [0, 300, 0].
 - [Width, Height] to about [750, 125].
 - In the Hierarchy tab, click on [Button_1 > Text] and in the Inspector tab, set:
 - [Text > Text] to read “**Full Size Scene**”.
 - [Text > Character > Font Size] to about [56].
- In the Hierarchy tab, right click on [Menu] and choose [UI > Button]. Name it “**Button_2a**” and in the Inspector tab, Set:
 - [Pos X, Pos Y, Pos Z] to about [-200, 100, 0].
 - [Width, Height] to about [350, 125].
 - In the Hierarchy tab, click on [Button_1 > Text] and in the Inspector tab, set:
 - [Text > Text] to read “**Small Model**”.
 - [Text > Character > Font Size] to about [56].
- In the Hierarchy tab, right click on [Menu] and choose [UI > Button]. Name it “**Button_2b**” and in the Inspector tab, Set:
 - [Pos X, Pos Y, Pos Z] to about [200, 100, 0].
 - [Width, Height] to about [350, 125].
 - In the Hierarchy tab, click on [Button_2 > Text] and in the Inspector tab, set:
 - [Text > Text] to read “**Reset Model**”.
 - [Text > Character > Font Size] to about [56].
- In the Hierarchy tab, right click on [Menu] and choose [UI > Toggle]. Name it “**Button_3**” and in the Inspector tab, Set:
 - [Pos X, Pos Y, Pos Z] to about [0, -100, 0].
 - [Width, Height] to about [750, 125].
 - In the Hierarchy tab, click on [Button_3 > Background] and in the Inspector tab, set:
 - [Pos X, Pos Y, Pos Z] to about [375, -62.5, 0].
 - [Width, Height] to about [750, 125].
 - In the Hierarchy tab, click on [Button_3 > Background > Checkmark] and in the Inspector tab, set:
 - [Image (Script)] to [OFF].
 - In the Hierarchy tab, click on [Button_3 > Label] and in the Inspector tab, set:
 - [Left, Top, Pos Z] to [0, 0, 0].
 - [Right, Bottom] to [0, 0].
 - [Text > Text] to read “**Audio - Play/Stop**”.
 - [Text > Character > Font Size] to about [56].
 - [Text > Paragraph > Alignment] to [Horizontal Centre] & [Vertical Centre].
- In the Hierarchy tab, right click on [Menu] and choose [UI > Button]. Name it “**Button_4**” and in the Inspector tab, Set:
 - [Pos X, Pos Y, Pos Z] to about [0, -300, 0].
 - [Width, Height] to about [750, 125].
 - In the Hierarchy tab, click on [Button_4 > Text] and in the Inspector tab, set:
 - [Text > Text] to read “**Exit Scene**”.
 - [Text > Character > Font Size] to about [56].

- In the Hierarchy tab, click on [SDK_Files > Script_Files > Controller_L and Controller_R] and in the Inspector tab:
 - Click on the [Add Component] button and search for “**VRTK_UIPointer**”.
 - Set [Activation Settings > Activation Button] to [Button Two Press].
(This is the top “Menu” button on the Vive controllers).
 - Set [Activation Settings > Activation Mode] to [Toggle Button].
 - Click on the [Add Component] button and search for “**VRTK_Pointer**”.
 - Set [Destination Marker Settings > Enable Teleport] to [OFF].
 - Set [Pointer Activation Settings > Activation Button] to [Button Two Press].
 - Set [Pointer Activation Settings > Hold Button To Activate] to [OFF].
 - Set [Pointer Selection Settings > Selection Button] to [Trigger Press].
 - Click on the [Add Component] button and search for “**VRTK_StraightPointerRenderer**”.
 - Set [General Appearance Settings > Valid Collision Color] to about [R:40, G:220, B:40, A:128].
 - Set [General Appearance Settings > Invalid Collision Color] to about [R:220, G:40, B:40, A:128].
 - Individually for Controller_L and then for Controller_R, in the Inspector tab, drag and drop [VRTK_Straight Pointer Renderer (Script)] onto [VRTK_Pointer > Pointer Activation Settings > Pointer Renderer].
- Drag and drop [Project > Scripts > **MenuToggle**] onto [Hierarchy tab > VRTK_Files > Script_Files > Controller_L].
- Drag and drop [Project > Scripts > **MenuToggle**] onto [Hierarchy tab > VRTK_Files > Script_Files > Controller_R].
- Click on [VRTK_Files > Script_Files > Controller_L]:
 - Drag and drop [VRTK_Files > Script_Files > Controller_L] onto [Inspector tab > Menu Toggle > Controller Events].
 - Drag and drop [Menu_Canvas > Menu_Follower > Menu] onto [Inspector tab > Menu Toggle > Menu].
- Click on [VRTK_Files > Script_Files > Controller_R]:
 - Drag and drop [VRTK_Files > Script_Files > Controller_R] onto [Inspector tab > Menu Toggle > Controller Events].
 - Drag and drop [Menu_Canvas > Menu_Follower > Menu] onto [Inspector tab > Menu Toggle > Menu].
- Click on [Menu_Canvas > Menu_Follower > Menu] and in the Inspector tab:
 - Turn OFF the display of the Menu object so that the Menu is OFF when the game starts.
- In the Hierarchy tab, click on [Menu_Canvas > Menu_Follower > Menu > **Button_1**] and in the Inspector tab:
 - Click on [Button > On Click > “+”].
 - From the Hierarchy tab, drag and drop [Your Game Object] to the [Button > On Click] item list.
 - Set [Button > On Click > Function List] to [Game Object > SetActive (bool)].
 - Set [Button > On Click > Check Box next to item list] to [ON] or [OFF].
 - Repeat for as many Game Objects that you need to turn ON or OFF..
- In the Hierarchy tab, click on [Menu_Canvas > Menu_Follower > Menu > **Button_2a**] and in the Inspector tab:
 - Click on [Button > On Click > “+”].
 - From the Hierarchy tab, drag and drop [Your Game Object] to the [Button > On Click] item list.
 - Set [Button > On Click > Function List] to [Game Object > SetActive (bool)].
 - Set [Button > On Click > Check Box next to item list] to [ON] or [OFF].
 - Repeat for as many Game Objects that you need to turn ON or OFF..
- Drag and drop [Project > Scripts > **ResetObjectPosition**] onto [Hierarchy tab > Your Grabbable Object].
(This is the Object that has the potential to be thrown out of reach).
- In the Hierarchy tab, click on [Menu_Canvas > Menu_Follower > Menu > **Button_2b**] and in the Inspector tab:
 - Click on [Button > On Click > “+”].
 - From the Hierarchy tab, drag and drop [Your Grabbable Object] to the [Button > On Click] item list.
 - Set [Button > On Click > Function List] to [ResetObjectPosition > resetTransform ()].
- In the Hierarchy tab, click on [Menu_Canvas > Menu_Follower > Menu > **Button_3**] and in the Inspector tab:
 - Click on [Button > On Click > “+”].
 - From the Hierarchy tab, drag and drop [Your Audio Object] to the [Button > On Click] item list.
 - Set [Button > On Click > Function List] to [AudioSource > enabled].
- Drag and drop [Project > Scripts > **ExitGame**] onto [Hierarchy tab > EventSystem].
- In the Hierarchy tab, click on [Menu_Canvas > Menu_Follower > Menu > **Button_4**] and in the Inspector tab:
 - Click on [Button > On Click > “+”].
 - From the Hierarchy tab, drag and drop [EventSystem] to the [Button > On Click] item list.
 - Set [Button > On Click > Function List] to [ExitGame > doExitGame ()].
- In the Hierarchy tab, remember to turn **ON** and **OFF** relevant Game Objects so that the Scene starts in the correct state.

View the Project with a VR Headset:

- Make sure that the Vive headset, base stations, and controllers are all turned on.
- Start SteamVR from your desktop.

Option 1 – Active project from within Unity:

- Open your VR project with Unity.
- Press the [Play] button at the top of the screen to load your project into VR.
- Press the [Play] button again to exit out of VR mode so that you can continue editing your project.

Option 2 – Build a stand alone executable:

- A stand alone executable is a self contained package of files that runs on any Windows computer.
- Goto [Toolbar > File > Build Settings...]
 - Click on the [Add Open Scenes] button to add the “Main” scene from your project.
 - In the Platform section, choose [**PC, Mac & Linux Standalone**].
 - Set [Target Platform] = [Windows] (Mac does not have the hardware needed to run VR applications)
 - Set [Architecture] = [x86_64]
 - Click on the [Build] button.
 - Choose a destination folder that is separate from your project files so that it is easier to isolate the self contained package of files that you need to give to anyone wishing to view the VR project.
- You should see an exe file, a dll file, and a Data folder.
- Using a file browser, double click on the exe file to play the VR application.
- Click on the ‘X’ button at the top right of the window to close the VR application.
- Put the Vive headset on to view the VR scene interactively.
- Walk around the scene as you would in real life.
- To travel further than the confines of your physical room, press the large “Touchpad” button on the controllers and aim the Bezier curve to the spot where you would like to be teleported.
- Use the “Grip” button to interact with grabbable objects.
- Use the “Menu” button to display the Pop-Up Menu.
- Use the “Trigger” button to interact with the Pop-Up Menu.

Known Workflow Difficulties:

- The FBX exporter does not export elements of type “Surface” to the Unity game engine. To fix this problem, I’ve gotten into the habit of using the “CONVERT BREP” command to convert all “Surface” elements to “Smart Surface” elements.
- Geometry that has a material that doesn’t use a texture map gets exported without UV texture coordinates. This generally won’t be a problem but it is something to be aware of.
- The FBX exporter appends the DGN file name to the material name. This leads to very long material names in the Unity editor but it also means that you get duplicate materials when you use multiple FBX files to export your project in smaller more manageable chunks.
- Unity’s Standard shader uses single sided rendering which means that surface normal directions are critical. MicroStation needs a more efficient method of show the direction of ALL surface normals at once.
- MicroStation’s “Mirror” tool makes a mess of all surface normals so use it as sparingly as possible.
- MicroStation’s “Change Surface Normal Direction” tool can be useful at times but it has an unfortunate tendency to rotate the UV texture coordinates on your geometry.
- You can see a more in depth discussion of these issues here:
<http://communities.bentley.com/products/microstation/f/273/t/120578>