

## Integrating Trivisio Colibri with Unity (via VRPN and OSVR)

Run VRPN server:

```
Terminal - mitya@mitya-linux: ~/bullds/Colibri/ColibriAPI-1.1.20160302-Linux-amd64/bln
File Edit View Terminal Tabs Help
mitya@mitya-linux:~/bullds/Colibri/ColibriAPI-1.1.20160302-Linux-amd64/bin$ ./vrpn_server 3884
ColibriAPI version: 1.1.20160302
2 Colibri sensors found
0: TIRK_S (FW 2.004)
1: SUITCCE (FW 1.010)
Network started...
```

osvr\_server\_config.json contents:

```
{
  "externalDevices": {
    "/myExternalDevice": {
      "deviceName": "Colibri",
      "server": "localhost:3884",
      "descriptor": {
        "interfaces": {
          "tracker": {
            "count": 2,
            "position": false,
            "orientation": true
          }
        }
      }
    }
  },
  "aliases": {
    "/me/head": {
      "rotate": {
        "axis": "x",
        "degrees": 0
      },
      "child": {
        "changeBasis": {
          "x": "y",
          "y": "-z",
          "z": "x"
        },
        "child": "/myExternalDevice/tracker/0"
      }
    },
    "/me/hand": {
      "child": {
        "changeBasis": {
          "x": "y",
          "y": "-z",
          "z": "x"
        },
        "child": "/myExternalDevice/tracker/1"
      }
    }
  }
}
```

Run OSVR server:

```
Terminal - mitya@mitya-linux: /media/DATA/Mitya/My_Projects/osvr/OSVR-Core/build/bin
File Edit View Terminal Tabs Help
mitya@mitya-linux:/media/DATA/Mitya/My_Projects/osvr/OSVR-Core/build/bin$ ./osvr_server
[OSVR Server] Using default config file - pass a filename on the command line to use a different one.
[OSVR Server] Using config file 'osvr_server_config.json'
[OSVR Server] Constructing server as configured...
[OSVR Server] Loading auto-loadable plugins...
[OSVR Server] Loading plugins...
[OSVR Server]
[OSVR Server] Instantiating configured drivers...
[OSVR Server]
[OSVR Server] External devices found and parsed from config file.
[OSVR Server] Aliases found and parsed from config file.
[OSVR Server] Using OSVR HDK for display configuration. Did not find an alternate valid 'display' object in config file.
[OSVR Server] Triggering automatic hardware detection...
[OSVR Server] Registering shutdown handler...
[OSVR Server] Starting server mainloop: OSVR Server is ready to go!
[OSVR] Performing hardware auto-detection.
[OSVR] Path tree updated or connection detected
[OSVR] Sending path tree to clients.
```

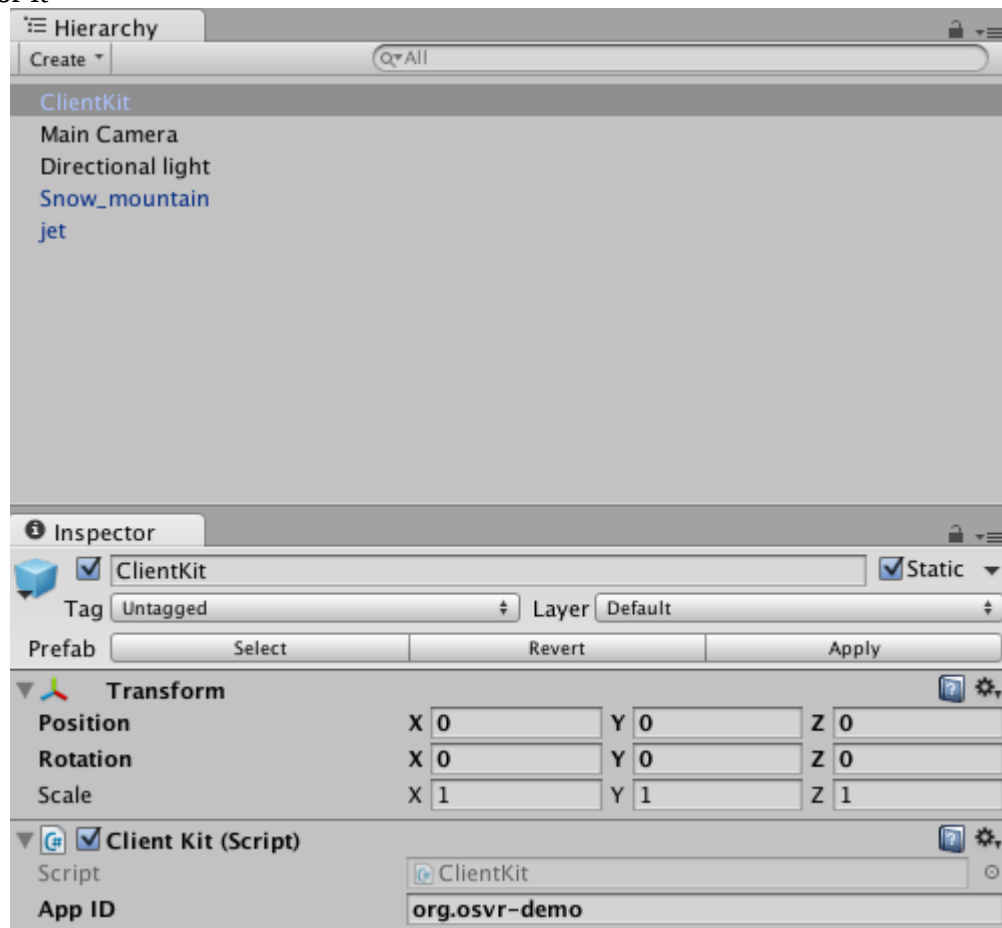
Get OSVR-Unity assets from <http://access.osvr.com/binary/osvr-unity>:

The screenshot shows a web browser window with the URL [access.osvr.com/binary/osvr-unity](http://access.osvr.com/binary/osvr-unity). The page features the OSVR logo and navigation links for 'Software Access', 'OSVR Developer Portal', and 'OSVR Home'. A call-to-action box prompts users to sign up for the Developer Newsletter. The main content area is titled 'OSVR-Unity Assets' and contains a detailed paragraph about the binary packages, including information about .NET Managed-OSVR binaries, C# source code, and native binaries for Windows. It also mentions a 'README.html' file and a '.unitypackage' file. Below the text, there is a table with the following data:

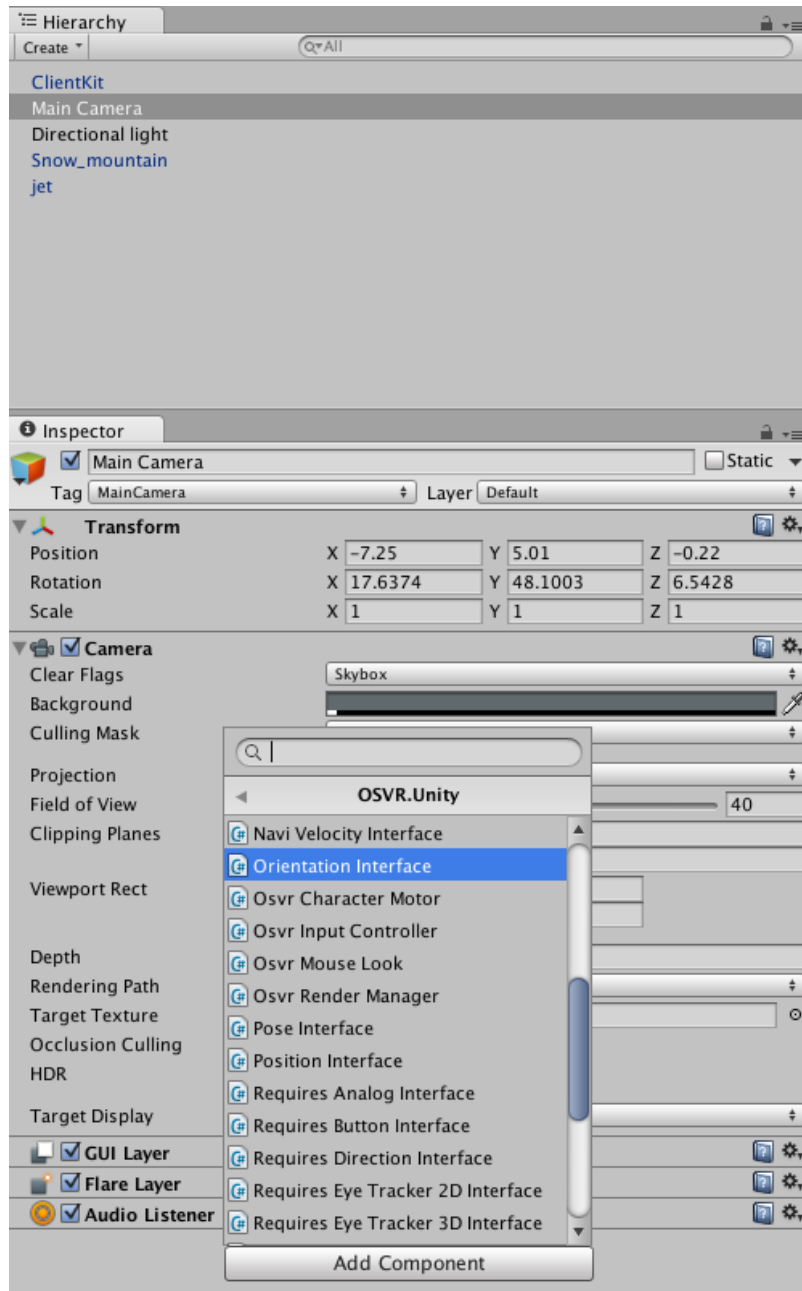
OSVR-Unity version	OSVR-Core version	Build number
0.6.8.0	0.6.1176-g9ba5951	360

In Unity:

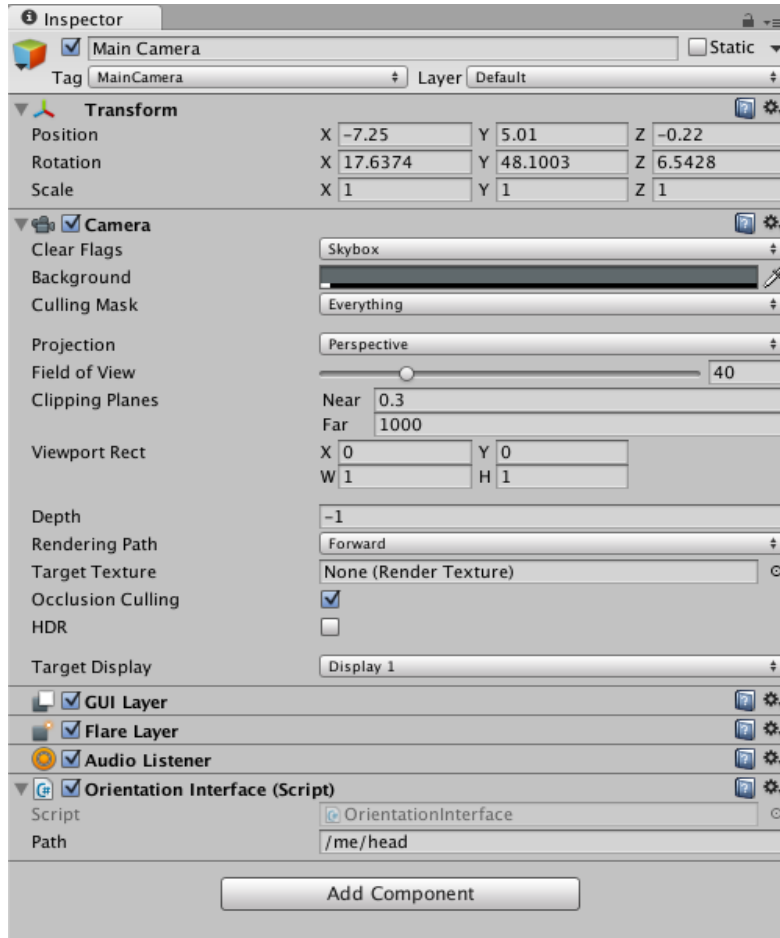
1. Create a new project
2. Project(tab)/Assets/Import New Asset...: choose OSVR-Unity.unitypackage, extracted from downloaded archive
3. Install OSVR-Core to Project\_Name/Assets/Plugins/Linux folder. What is really needed is only:  
libosvrClient.so, ...so.0, ...so.0.6  
libosvrClientKit.so ...so.0, ...so.0.6  
libosvrCommon.so ...so.0, ...so.0.6  
libosvrUtil.so ...so.0, ...so.0.6
4. Rename libosvrClientKit.so to osvClientKit.so
5. Drag and drop Assets/OSVRUnity/Prefabs/ClientKit to project's Hierarchy(tab) and set an App ID for it



6. For Main\_Camera, or any other object do  
Add\_Component/Scripts/OSVR.Unity/Orientation\_Interface



7. Set OSVR path to your tracker, for example /me/head



More information about Integrating OSVR with Unity is here: <http://osvr.github.io/doc/unity/>