sensics

OSVR

Using OSVR to Support Any Device in VR/AR

Yuval Boger

CEO/VRguy, Sensics

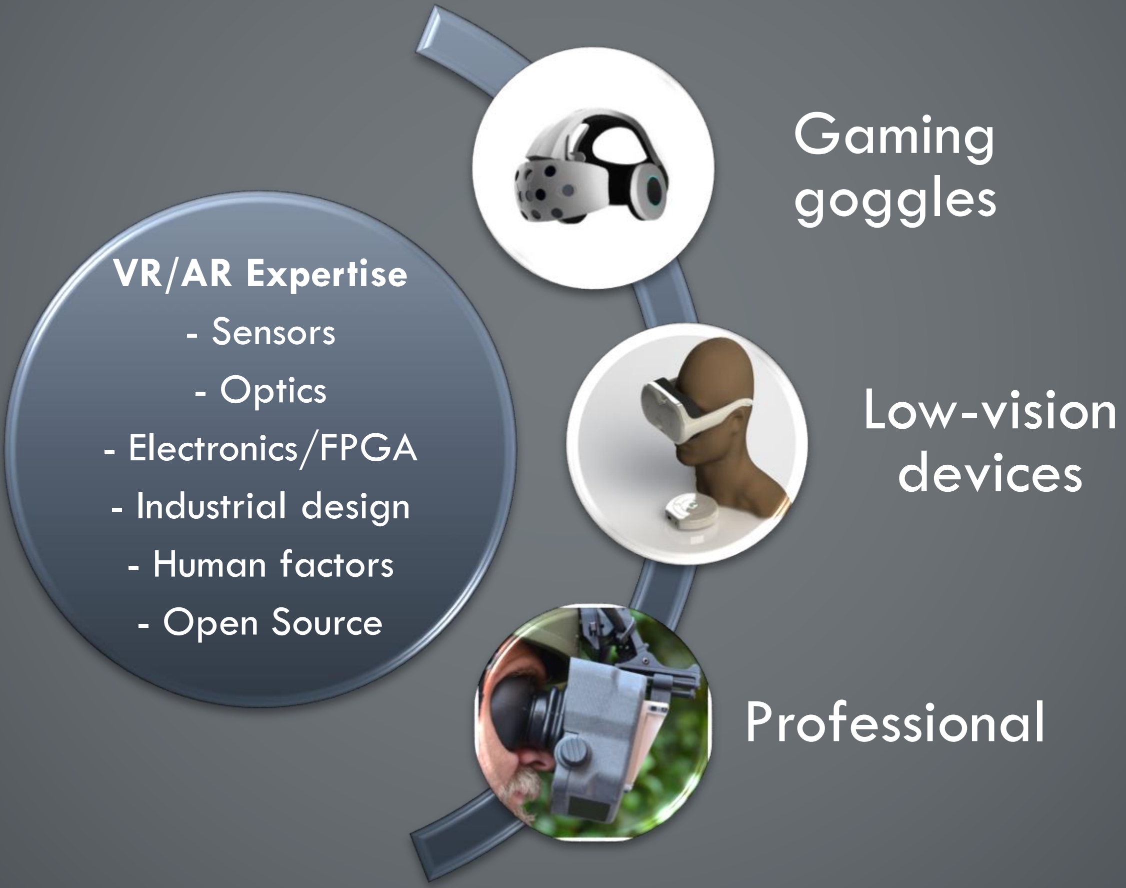
vrguy@sensics.com

VISION SUMMIT 2016

What if your Web Browser **only worked** with a Logitech Wireless Mouse?

No one wants to write a game that works only on one device

Sensics Experience



VR/AR Expertise

- Sensors
- Optics
- Electronics/FPGA
- Industrial design
- Human factors
- Open Source

Gaming goggles

Low-vision devices

Professional

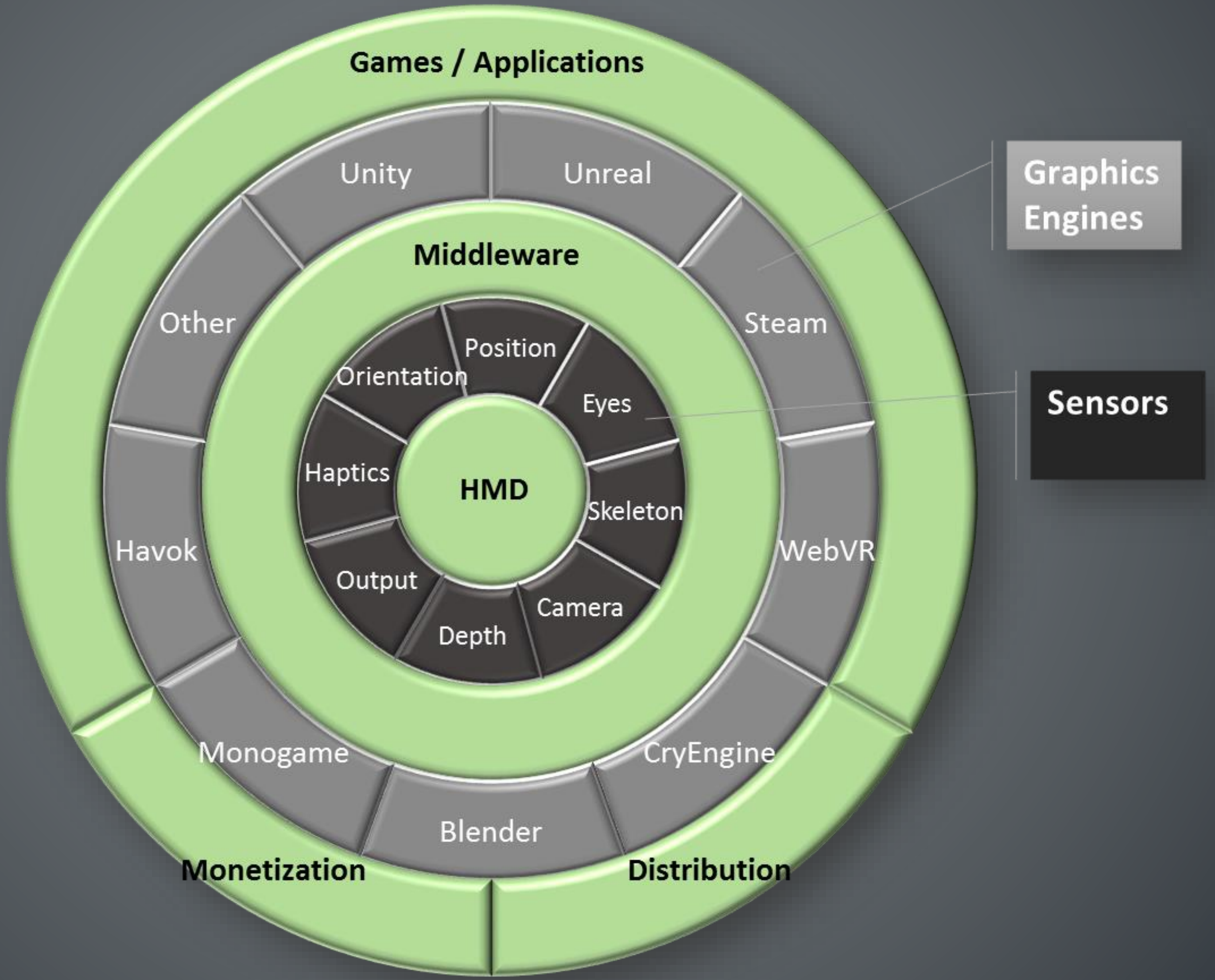
Gaming Goggles

Two Independent Parts of OSVR

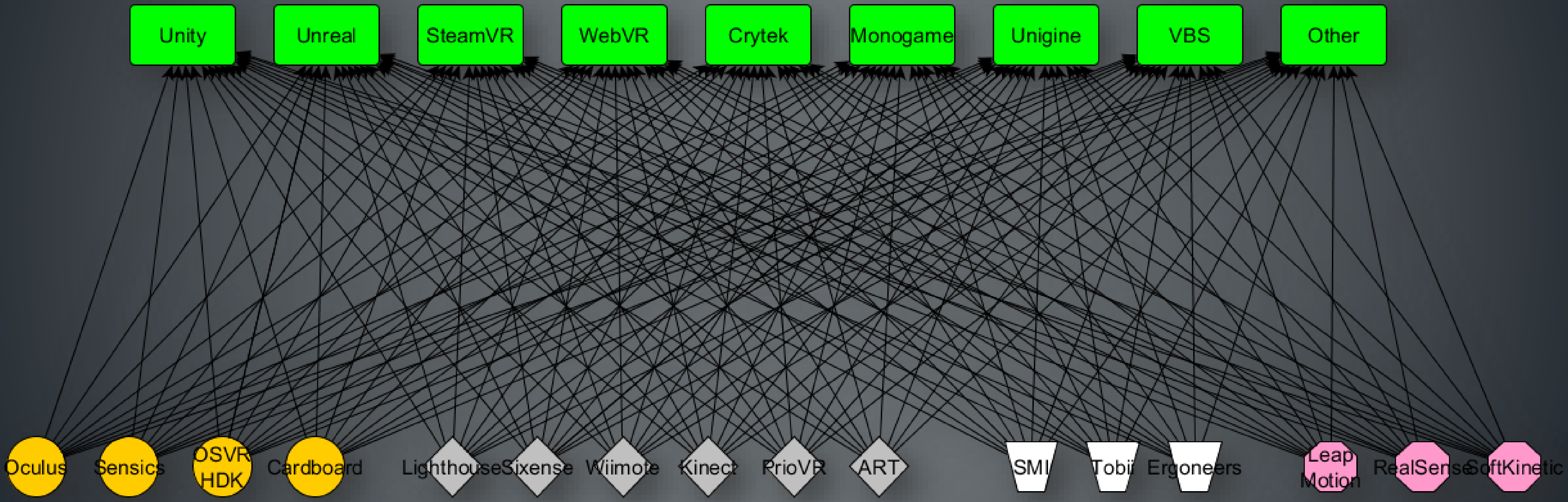
Open-source Hardware

Open-Source Software

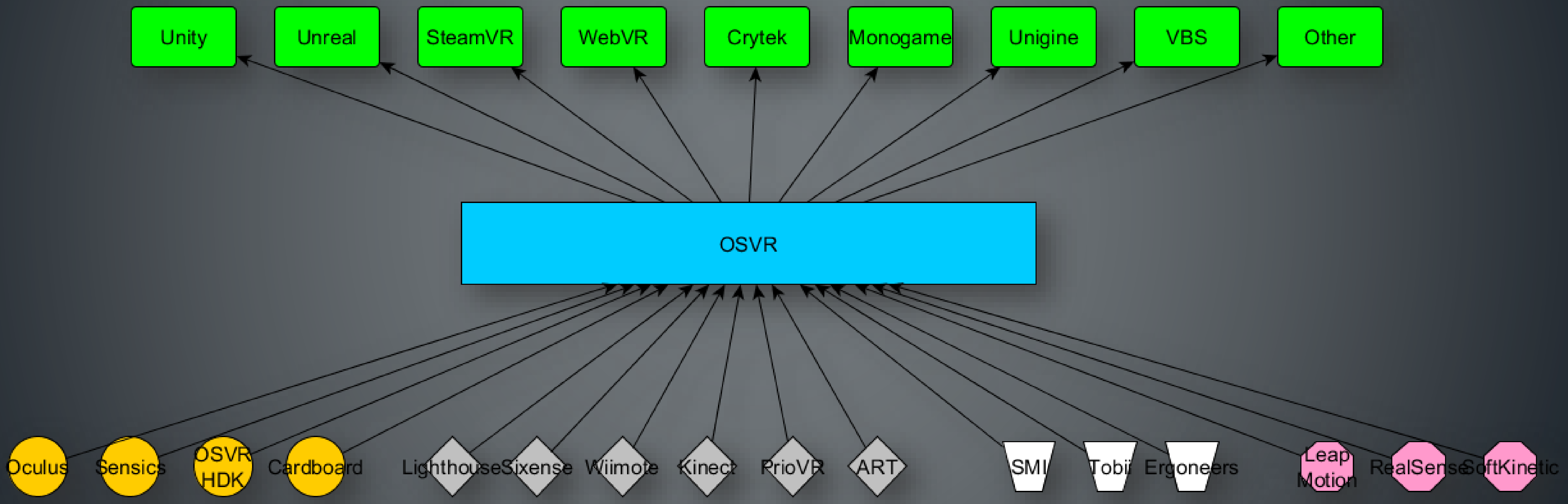
The VR Stack


Games / Applications
Unity
Unreal
Steam
WebVR
CryEngine
Blender
Monogame
Havok
Other
Middleware
Position
Eyes
Skeleton
Camera
Depth
Output
Haptics
Orientation
HMD
Monetization
Distribution
Graphics Engines
Sensors

Without OSVR



With OSVR

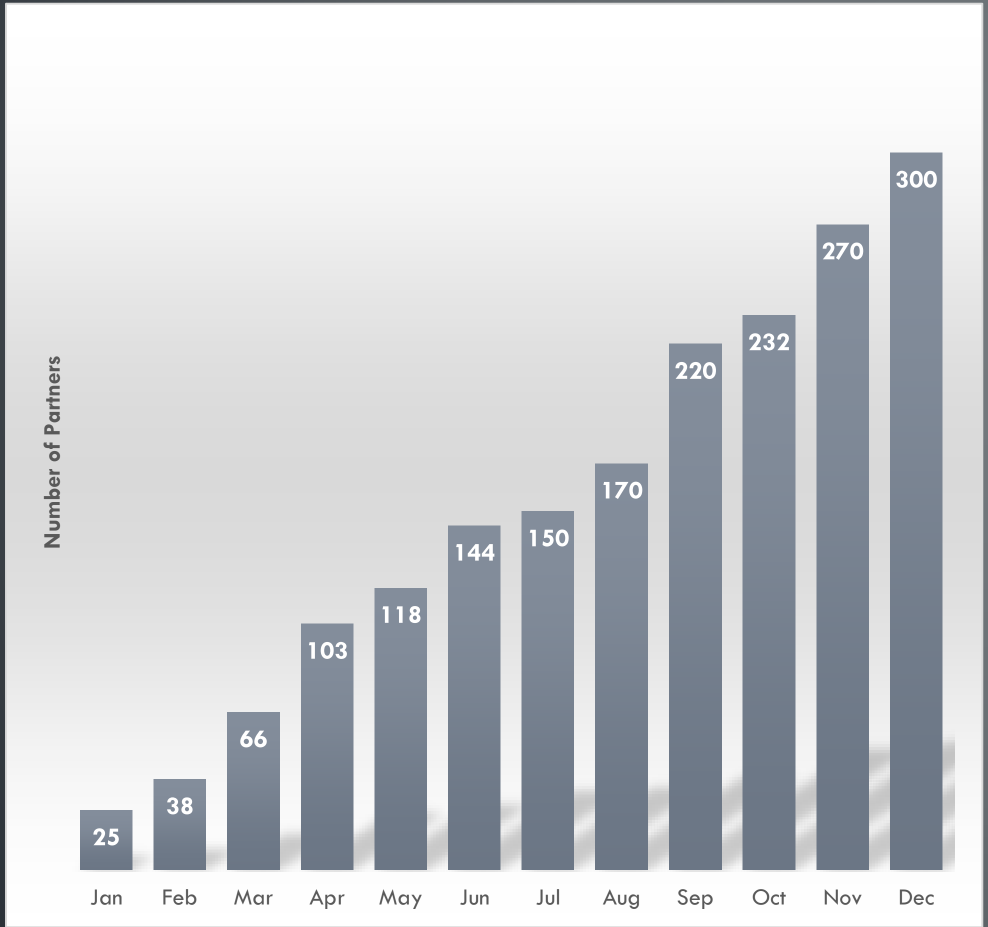

Unity
Unreal
SteamVR
WebVR
Crytek
Monogame
Unigine
VBS
Other
OSVR
Oculus
Sensics
OSVR HDK
Cardboard
Lighthouse
Sixense
Wiimote
Kinect
PrioVR
ART
SMI
Tobii
Ergoneers
Leap Motion
RealSense
SoftKinetic

What Is the OSVR™ Software?

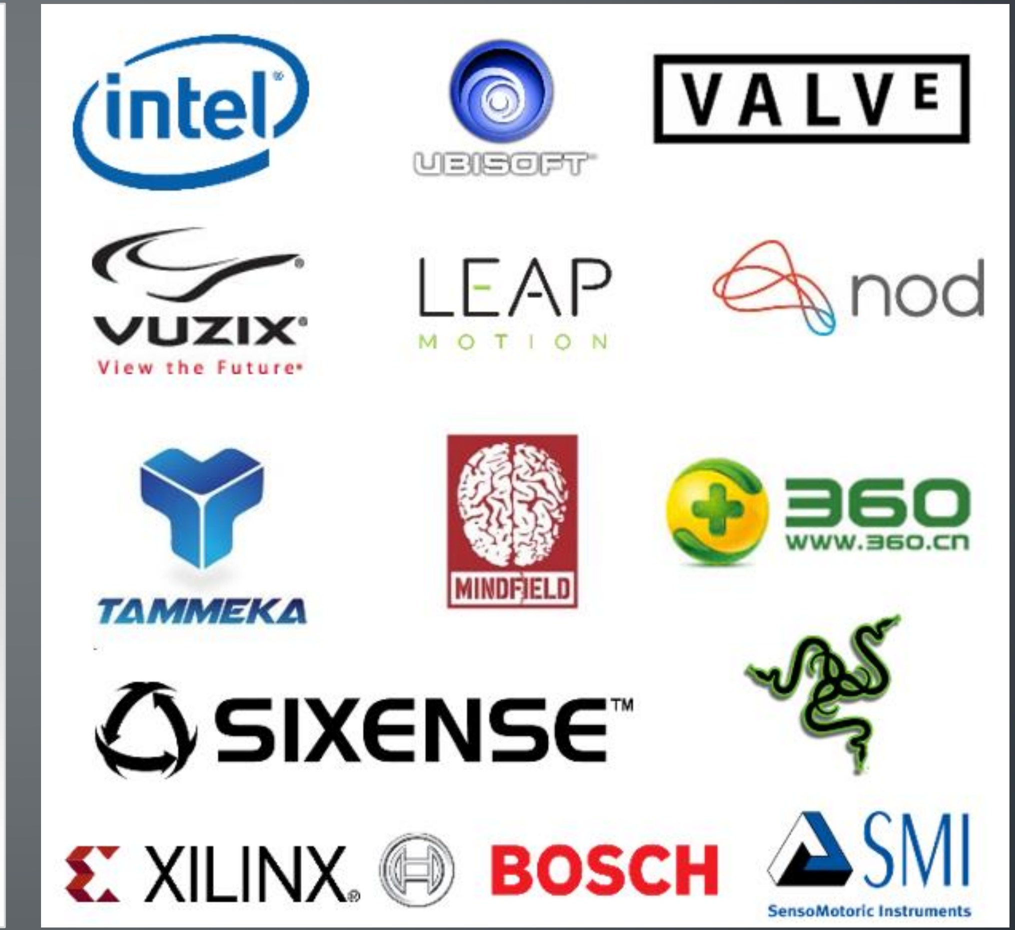
- A multi-platform, standardized interface to virtual reality devices and peripherals.
- A set of high-performance rendering utilities
- Highly extensible
- Free and open source, Apache 2.0 license

Growth of the OSVR Ecosystem

Number of Participants

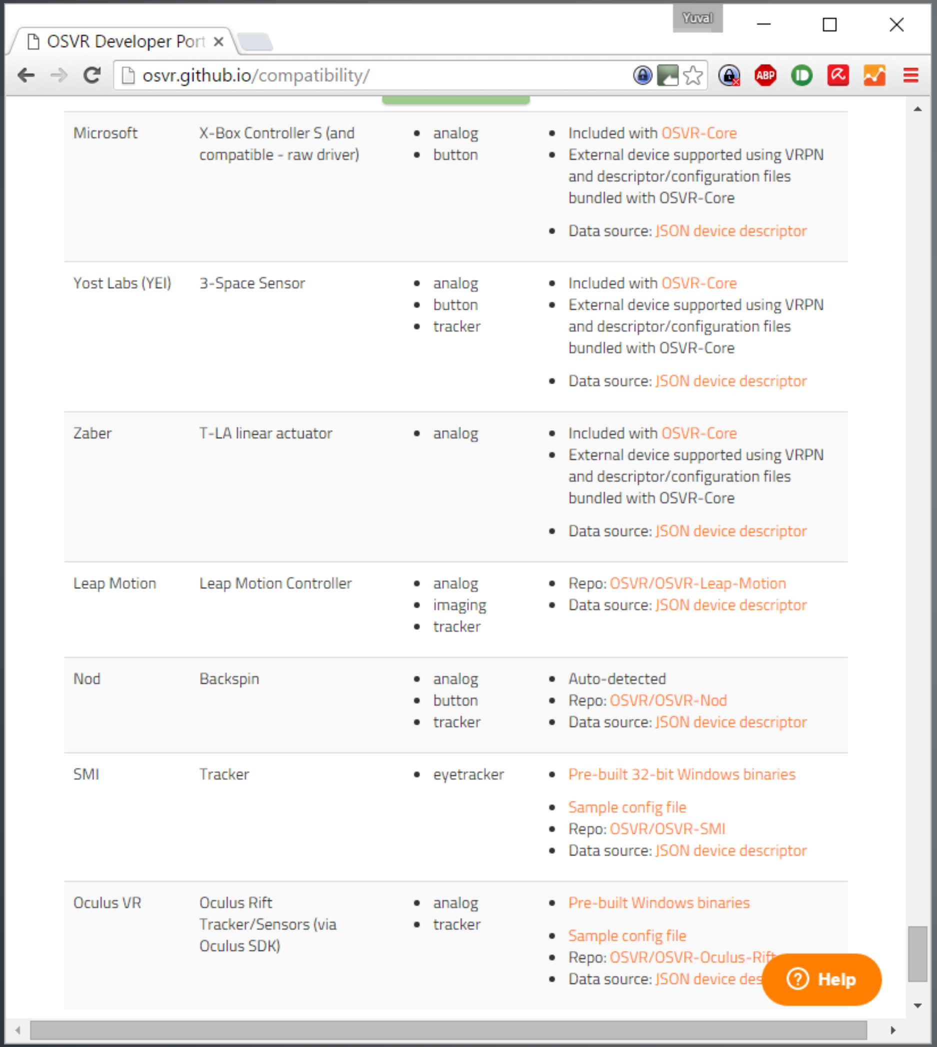


Selected Participants



Device and Platform Support

Hundreds of devices (see osvr.github.io)



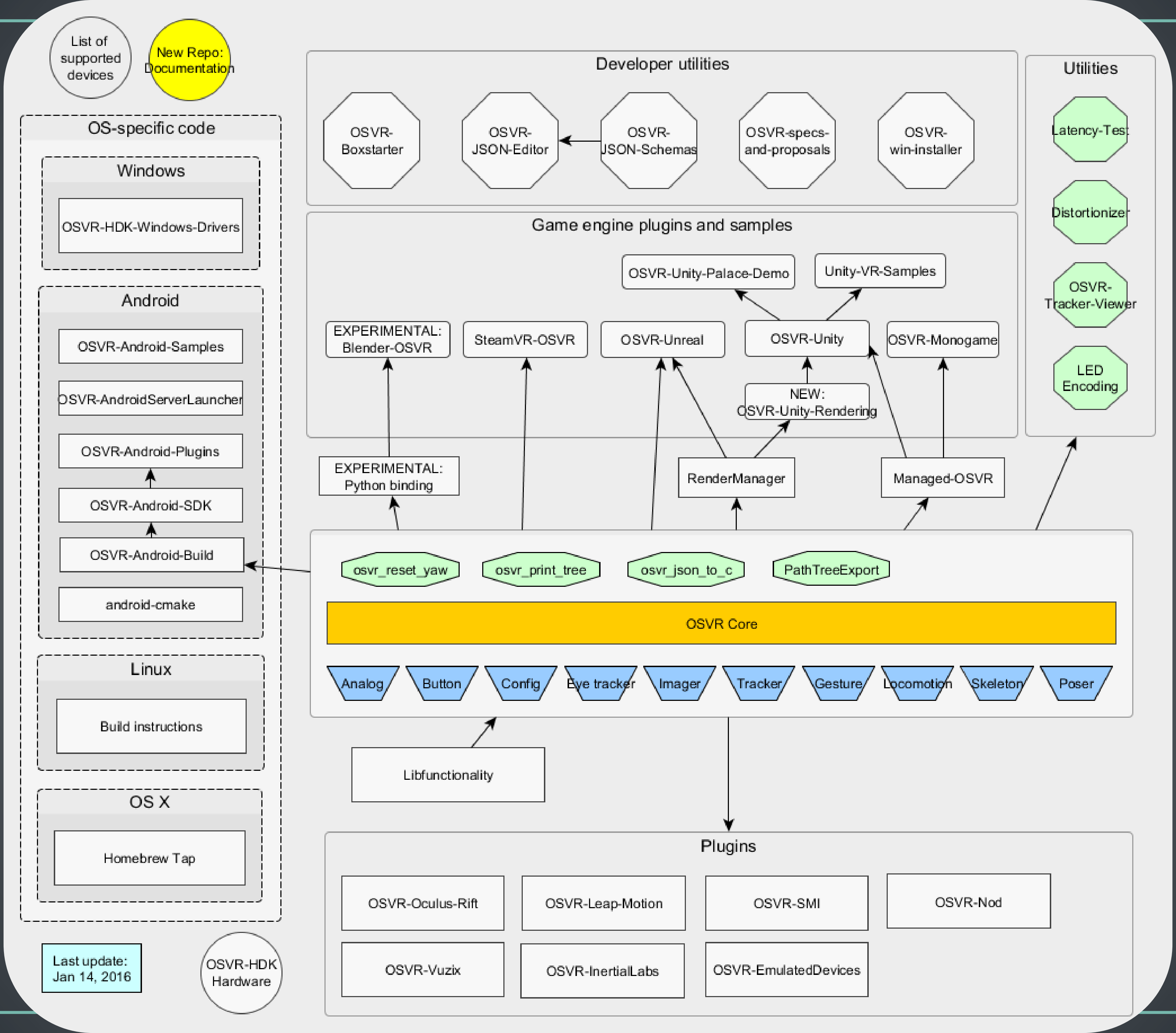
Microsoft	X-Box Controller S (and compatible - raw driver)	analog button	Included with OSVR-Core External device supported using VRPN and descriptor/configuration files bundled with OSVR-Core Data source: JSON device descriptor
Yost Labs (YEI)	3-Space Sensor	analog button tracker	Included with OSVR-Core External device supported using VRPN and descriptor/configuration files bundled with OSVR-Core Data source: JSON device descriptor
Zaber	T-LA linear actuator	analog	Included with OSVR-Core External device supported using VRPN and descriptor/configuration files bundled with OSVR-Core Data source: JSON device descriptor
Leap Motion	Leap Motion Controller	analog imaging tracker	Repo: OSVR/OSVR-Leap-Motion Data source: JSON device descriptor
Nod	Backspin	analog button tracker	Auto-detected Repo: OSVR/OSVR-Nod Data source: JSON device descriptor
SMI	Tracker	eyetracker	Pre-built 32-bit Windows binaries Sample config file Repo: OSVR/OSVR-SMI Data source: JSON device descriptor
Oculus VR	Oculus Rift Tracker/Sensors (via Oculus SDK)	analog tracker	Pre-built Windows binaries Sample config file Repo: OSVR/OSVR-Oculus-Rif Data source: JSON device des

Game Engines

- Unity
- Unreal
- CryEngine
- WebVR
- SteamVR
- Blender
- Monogame
- DirectX
- OpenGL
- And more

Windows

OSX



List of supported devices

New Repo: Documentation

OS-specific code

Windows

OSVR-HDK-Windows-Drivers

Android

OSVR-Android-Samples

OSVR-AndroidServerLauncher

OSVR-Android-Plugins

OSVR-Android-SDK

OSVR-Android-Build

android-cmake

Linux

Build instructions

OS X

Homebrew Tap

Last update: Jan 14, 2016

OSVR-HDK Hardware

Developer utilities

OSVR-Boxstarter

OSVR-JSON-Editor

OSVR-JSON-Schemas

OSVR-specs-and-proposals

OSVR-win-installer

Utilities

Latency-Test

Distortionizer

OSVR-Tracker-Viewer

LED Encoding

Game engine plugins and samples

OSVR-Unity-Palace-Demo

Unity-VR-Samples

EXPERIMENTAL: Blender-OSVR

SteamVR-OSVR

OSVR-Unreal

OSVR-Unity

OSVR-Monogame

NEW: OSVR-Unity-Rendering

EXPERIMENTAL: Python binding

RenderManager

Managed-OSVR

osvr_reset_yaw

osvr_print_tree

osvr_json_to_c

PathTreeExport

OSVR Core

Analog

Button

Config

Eye tracker

Imager

Tracker

Gesture

Locomotion

Skeleton

Poser

Libfunctionality

Plugins

OSVR-Oculus-Rift

OSVR-Leap-Motion

OSVR-SMI

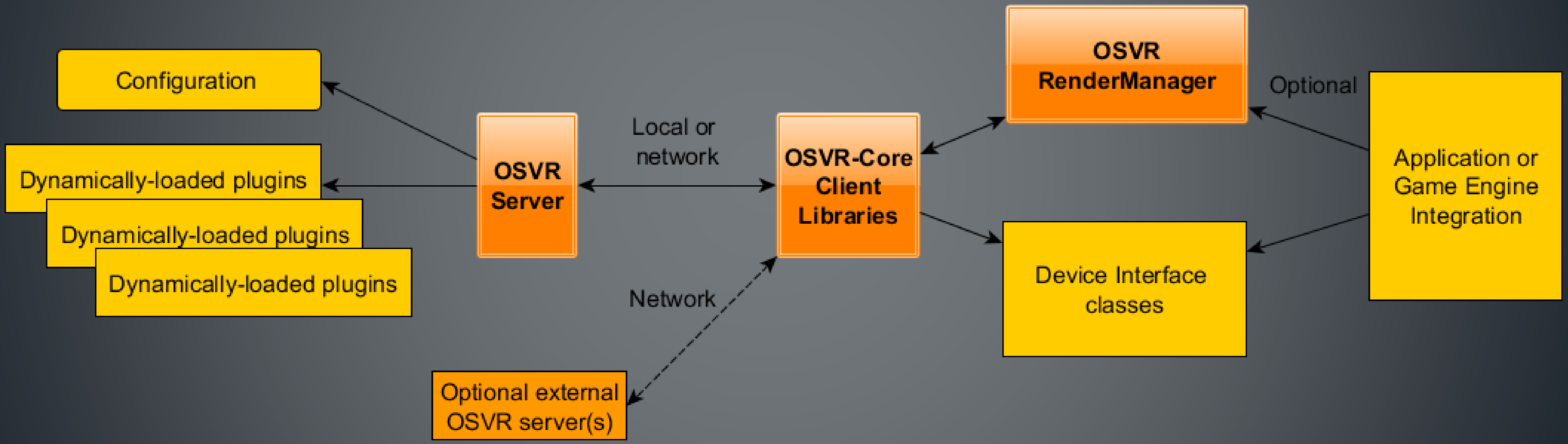
OSVR-Nod

OSVR-Vuzix

OSVR-InertialLabs

OSVR-EmulatedDevices

OSVR Block Diagram


Configuration
Dynamically-loaded plugins
Dynamically-loaded plugins
Dynamically-loaded plugins
OSVR Server
Local or network
OSVR-Core Client Libraries
Network
Optional external OSVR server(s)
OSVR RenderManager
Optional
Device Interface classes
Application or Game Engine Integration

OSVR Interfaces

- **Interfaces** are the primitive "pipes of data“
- A *Device* exposes one or more *interfaces*

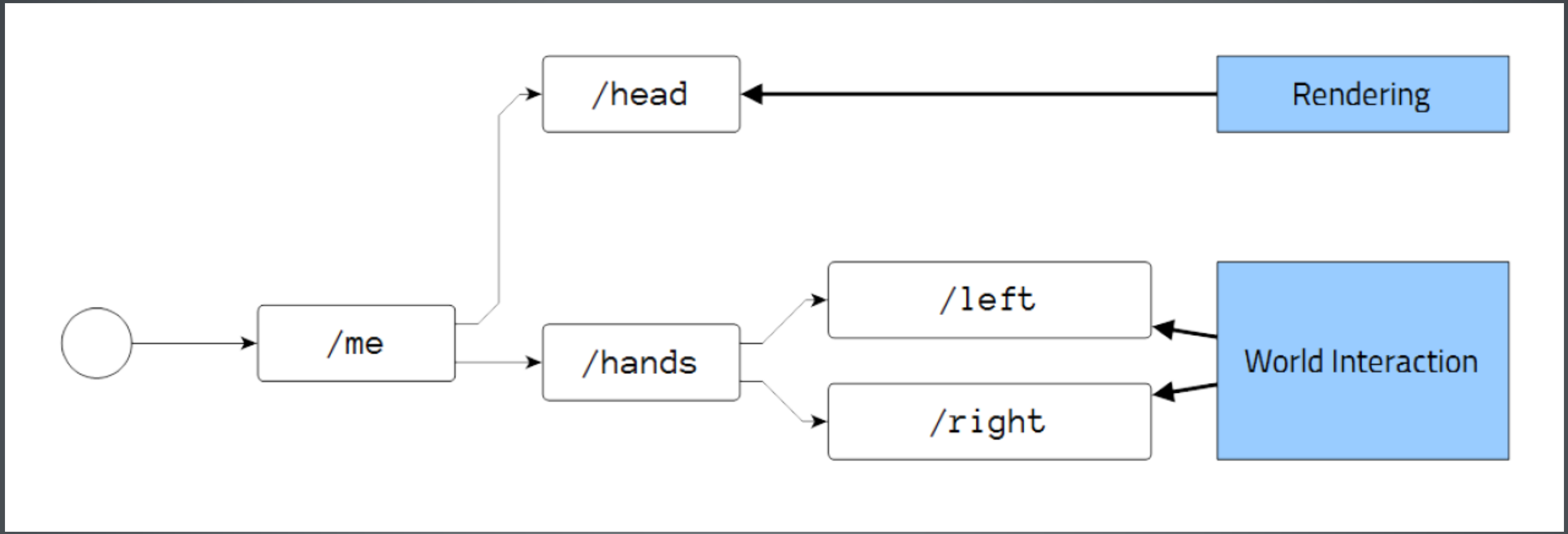
- Tracker
- Eye tracking
- Display
- Imager (camera)
- Skeleton (e.g. hand, full body)
- Analog
- Button
- Gesture
- Poser (e.g. motion platforms)

Two Ways of working with an Interface

Synchronous: blocking read or write

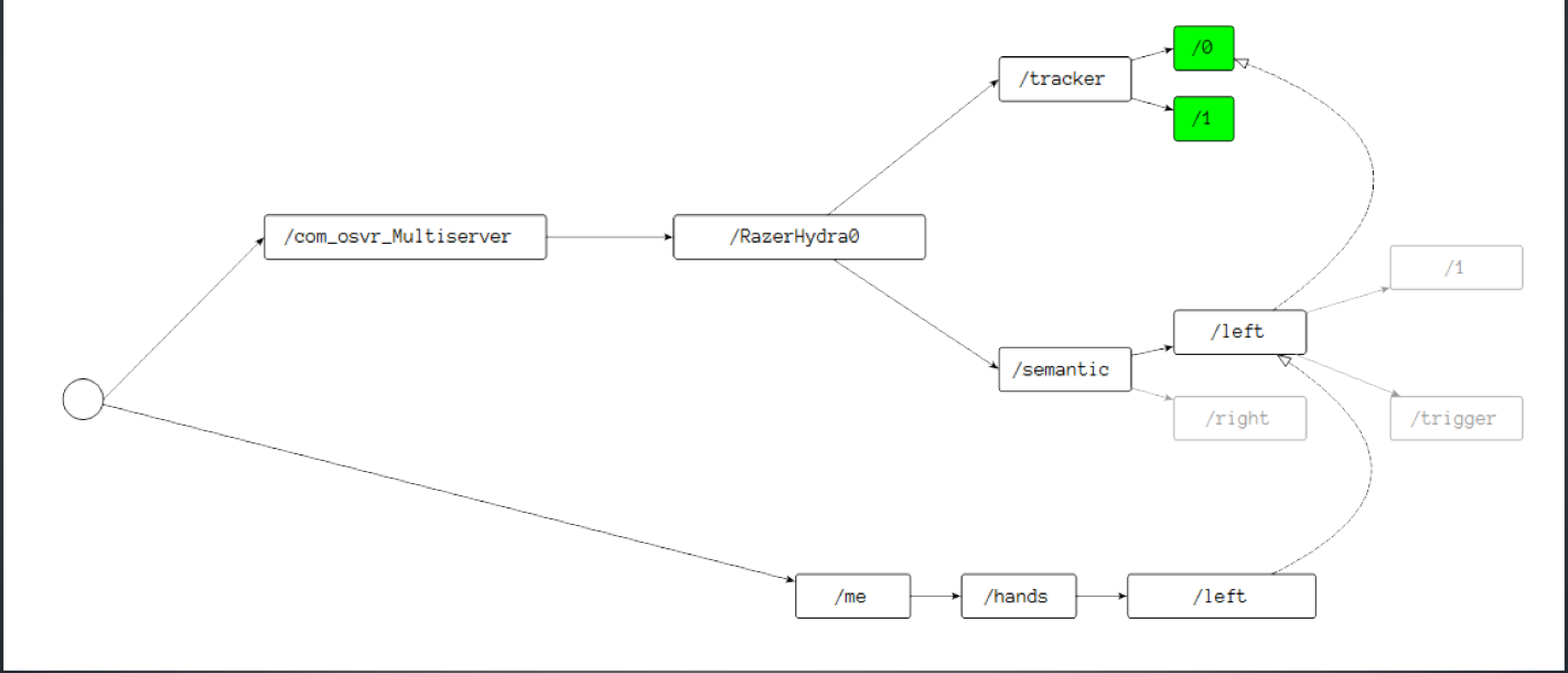
Asynchronous: callback for an event

An App Uses a Semantic Path



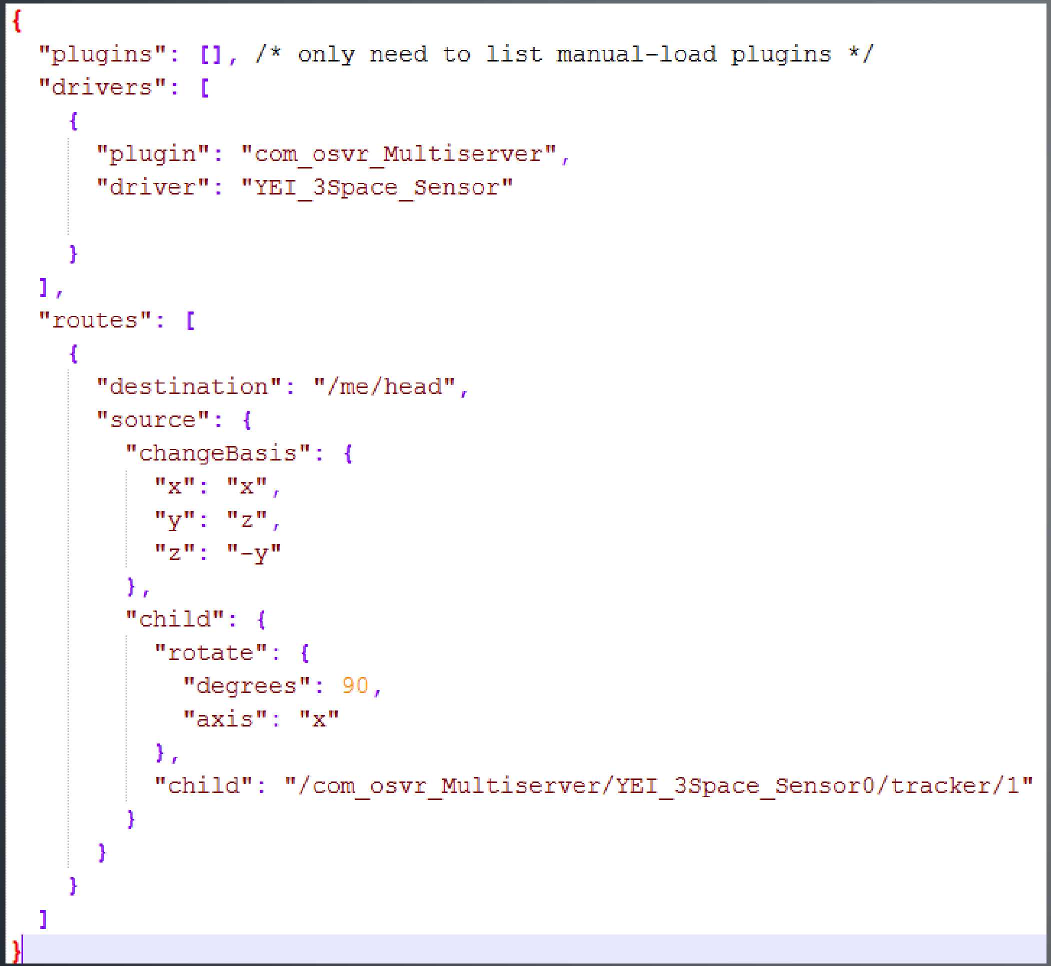
```
osvr::clientkit::Interface lefthand = context.getInterface("/me/hands/left");


OSVR_ReturnCode ret =  osvrGetPoseState(lefthand.get(), &timestamp, &state);
                                   or
osvrRegisterPoseCallback(lefthand, &myTrackerCallback, NULL);
```

...Allowing Mapping to a Device



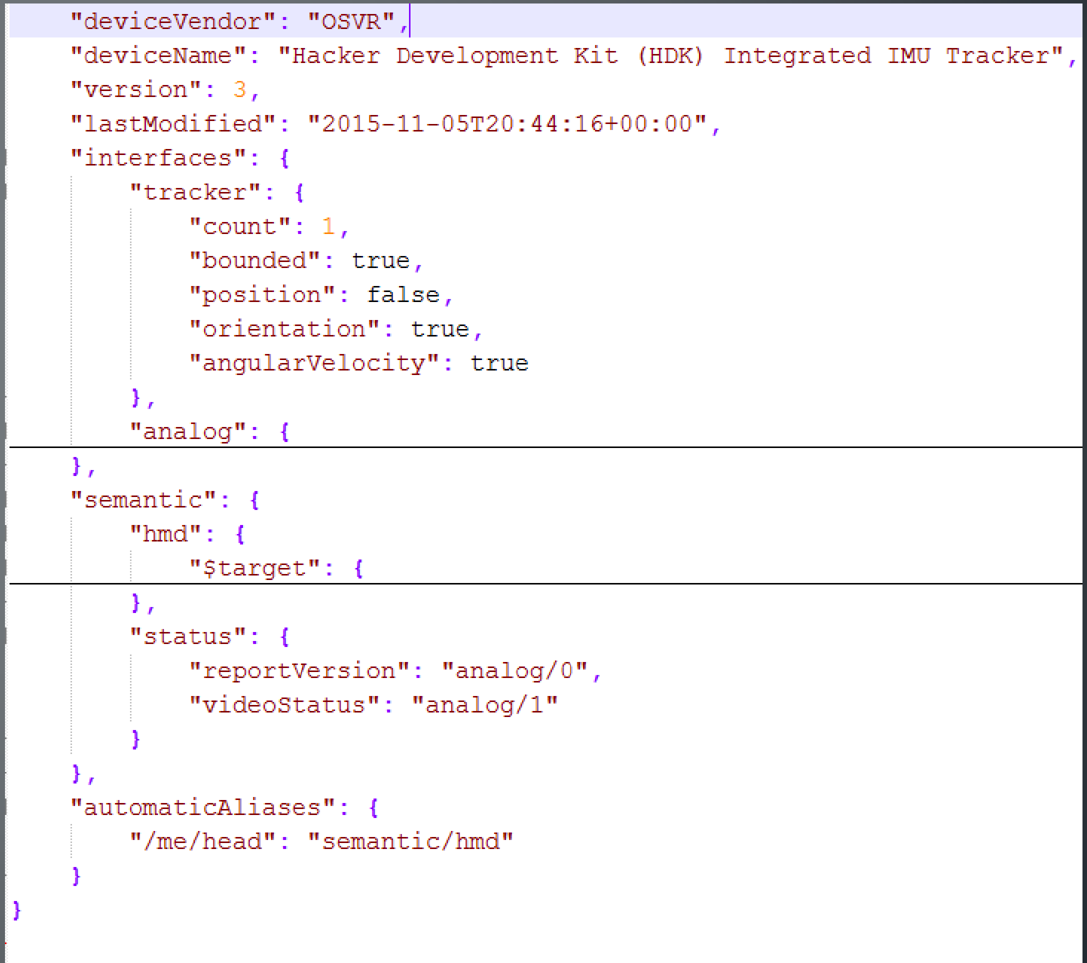
...at Runtime!

YEI Tracker



```
{
  "plugins": [], /* only need to list manual-load plugins */
  "drivers": [
    {
      "plugin": "com_osvr_Multiserver",
      "driver": "YEI_3Space_Sensor"

    }
  ],
  "routes": [
    {
      "destination": "/me/head",
      "source": {
        "changeBasis": {
          "x": "x",
          "y": "z",
          "z": "-y"
        },
        "child": {
          "rotate": {
            "degrees": 90,
            "axis": "x"
          },
          "child": "/com_osvr_Multiserver/YEI_3Space_Sensor0/tracker/1"
        }
      }
    }
  ]
}
```

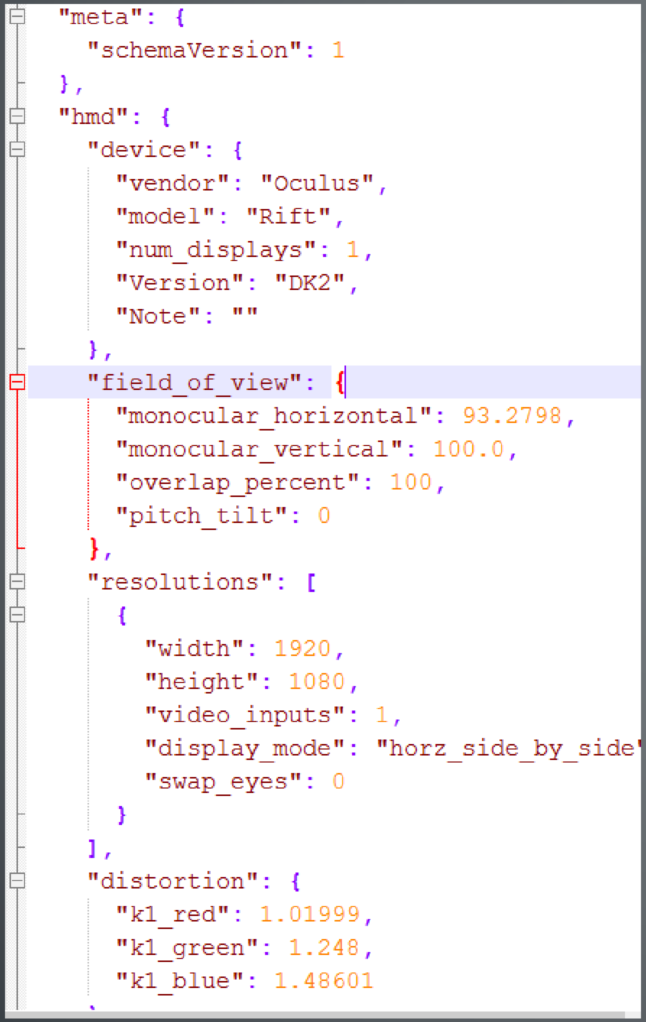
OSVR HDK Tracker



```
    "deviceVendor": "OSVR",
    "deviceName": "Hacker Development Kit (HDK) Integrated IMU Tracker",
    "version": 3,
    "lastModified": "2015-11-05T20:44:16+00:00",
    "interfaces": {
        "tracker": {
            "count": 1,
            "bounded": true,
            "position": false,
            "orientation": true,
            "angularVelocity": true
        },
        "analog": {
    },
    "semantic": {
        "hmd": {
            "$target": {
        },
        "status": {
            "reportVersion": "analog/0",
            "videoStatus": "analog/1"
        }
    },
    "automaticAliases": {
        "/me/head": "semantic/hmd"
    }
}
```

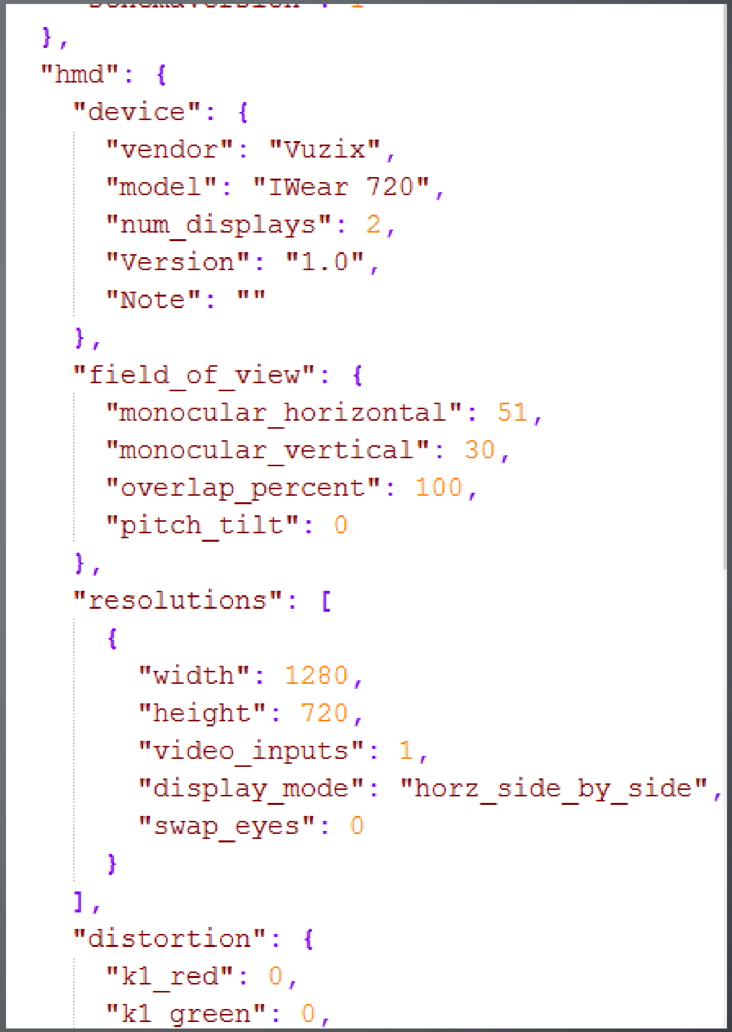
Same is True for Displays

DK2



```
"meta": {
  "schemaVersion": 1
},
"hmd": {
  "device": {
    "vendor": "Oculus",
    "model": "Rift",
    "num_displays": 1,
    "Version": "DK2",
    "Note": ""
  },
  "field_of_view": {
    "monocular_horizontal": 93.2798,
    "monocular_vertical": 100.0,
    "overlap_percent": 100,
    "pitch_tilt": 0
  },
  "resolutions": [
    {
      "width": 1920,
      "height": 1080,
      "video_inputs": 1,
      "display_mode": "horz_side_by_side"
      "swap_eyes": 0
    }
  ],
  "distortion": {
    "k1_red": 1.01999,
    "k1_green": 1.248,
    "k1_blue": 1.48601
```

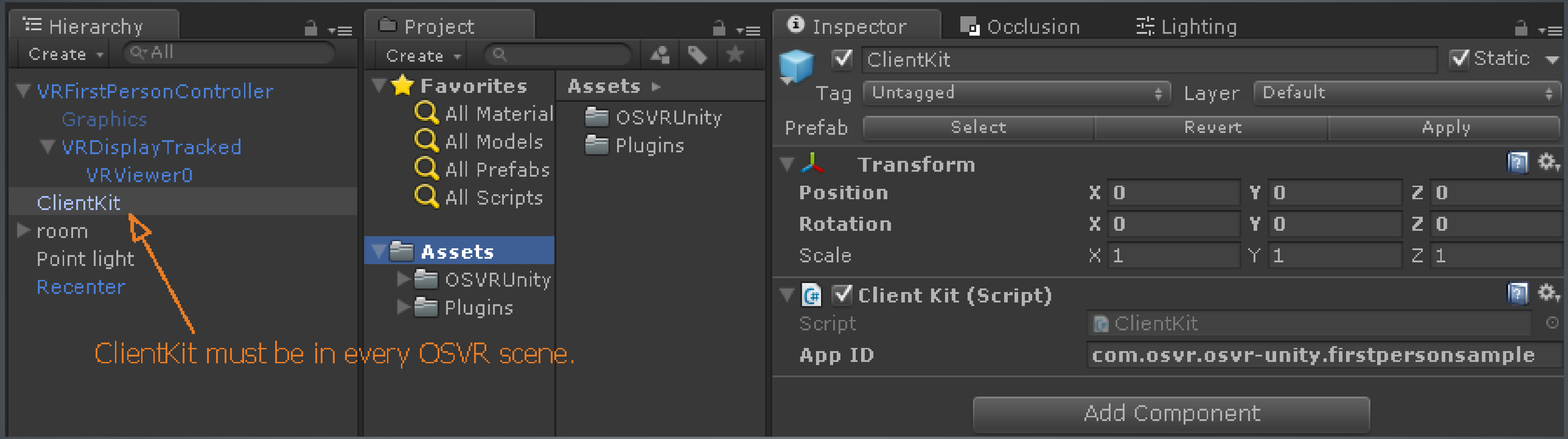
Vuzix 720



```
  "schemaVersion": 1
},
"hmd": {
  "device": {
    "vendor": "Vuzix",
    "model": "IWear 720",
    "num_displays": 2,
    "Version": "1.0",
    "Note": ""
  },
  "field_of_view": {
    "monocular_horizontal": 51,
    "monocular_vertical": 30,
    "overlap_percent": 100,
    "pitch_tilt": 0
  },
  "resolutions": [
    {
      "width": 1280,
      "height": 720,
      "video_inputs": 1,
      "display_mode": "horz_side_by_side",
      "swap_eyes": 0
    }
  ],
  "distortion": {
    "k1_red": 0,
    "k1_green": 0,
```

Unity Integration

https://github.com/OSVR/OSVR-Docs



VRFirstPersonController
Graphics
VRDisplayTracked
VRViewer0
ClientKit
room
Point light
Recenter

Favorites
All Material
All Models
All Prefabs
All Scripts
Assets
OSVRUnity
Plugins

Assets ▸
OSVRUnity
Plugins

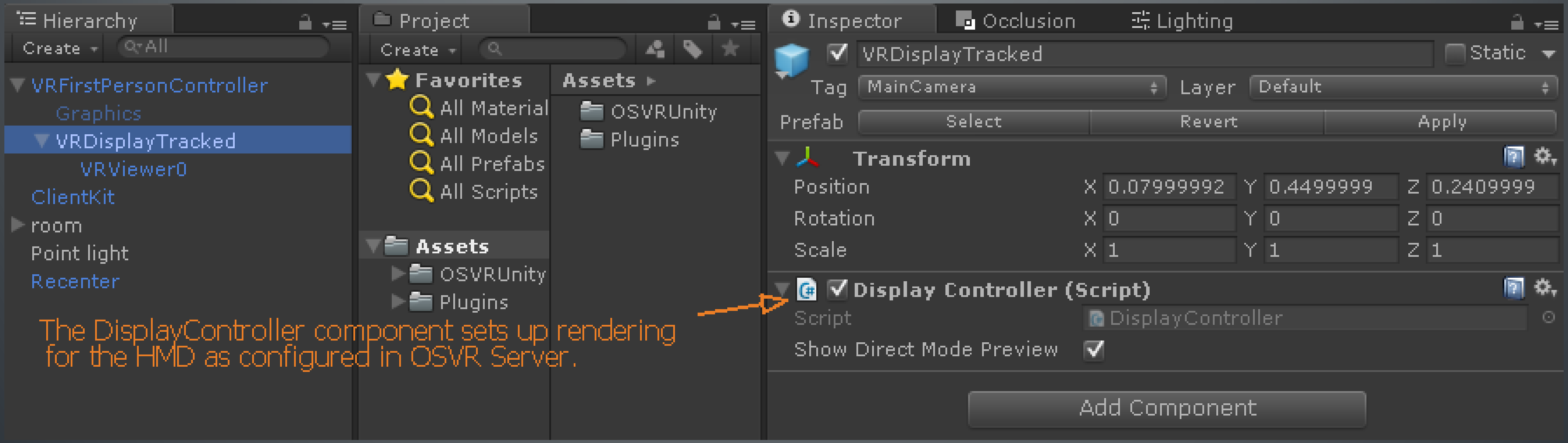
☑ ClientKit ☑ Static
Tag Untagged Layer Default
Prefab

Transform

Position	X 0	Y 0	Z 0
Rotation	X 0	Y 0	Z 0
Scale	X 1	Y 1	Z 1

☑ Client Kit (Script)
Script ClientKit
App ID com.osvr.osvr-unity.firstpersonsample

ClientKit must be in every OSVR scene.



VRFirstPersonController
Graphics
VRDisplayTracked
VRViewer0
ClientKit
room
Point light
Recenter

Favorites
All Material
All Models
All Prefabs
All Scripts
Assets
OSVRUnity
Plugins

Assets ▸
OSVRUnity
Plugins

☑ VRDisplayTracked □ Static
Tag MainCamera Layer Default
Prefab

Transform

Position	X 0.07999992	Y 0.4499999	Z 0.2409999
Rotation	X 0	Y 0	Z 0
Scale	X 1	Y 1	Z 1

☑ Display Controller (Script)
Script DisplayController
Show Direct Mode Preview ☑

The DisplayController component sets up rendering for the HMD as configured in OSVR Server.

OSVR Render Manager

- Direct mode
- Asynchronous time warp
- Distortion correction
- Preview window

- Easy to add new or custom HMDs
- Next up: foveated rendering

OSVR Plugins

- Dynamically loaded, open- or closed-source
- Device plugins
- Analysis plugins, such as:
  - Sensor fusion
  - Predictive tracking
  - Position detection
  - Augmented reality

Some Ongoing and Future Work

- Additional devices, platforms and game engine integrations
- Graphical tools to make configuration easier
- Foveated rendering
- New analysis plugins such as gesture engine, augmented reality

- It’s free and open – so we hope to see your code as well!

Summary of OSVR Advantages

- Supports multiple devices, operating systems, game engines
- Unified, device-independent programming model
- Optimized game engine interfaces
- Full set of capabilities
- Free and open source

- For additional information:
  - Yuval Boger, vrguy@sensics.com
  - osvr.github.io ; www.osvr.org

sensics

OS VR OSVR