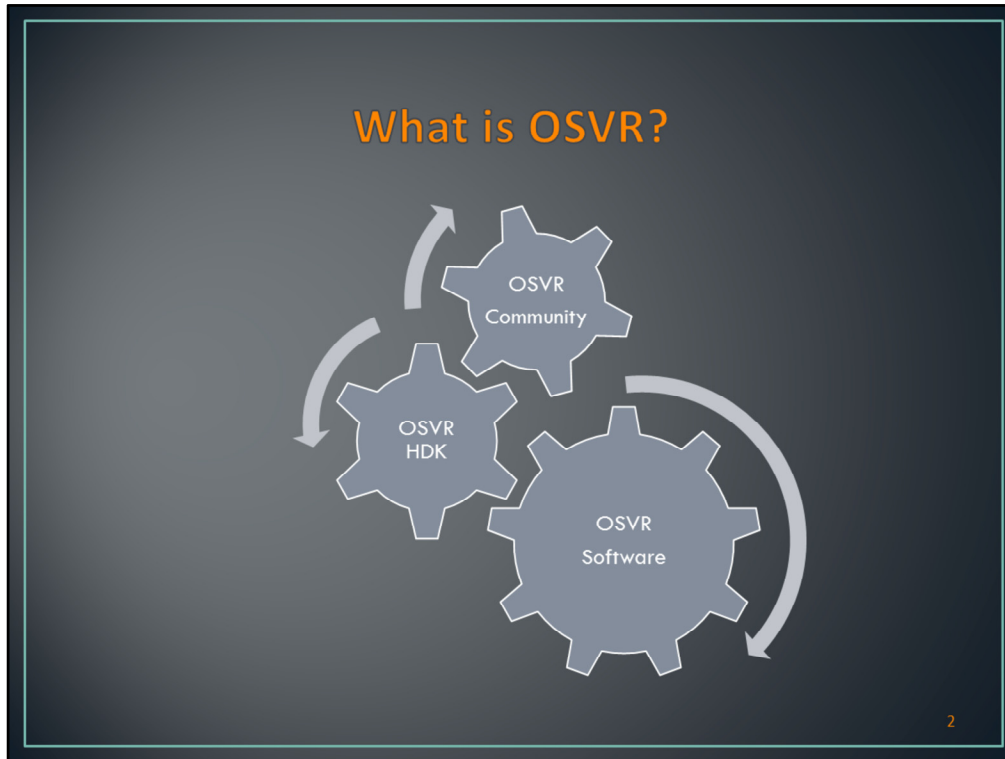




This presentation provides an over of OSVR – the open source virtual reality project.

It was originally presented at a ConVRge on-line event in April 2015



OSVR has three key parts:

1. The OSVR “hacker developer kit” (or HDK for short) which is a wide-field, open-source head-mounted displays that provides excellent image quality, 1080p screen and a host of innovative features. The HDK will soon be available for purchase for about \$200 at www.osvr.com
2. The OSVR software, a free and open-source multi-platform software framework that allows standardized discovery, configuration and operation of a wide range of VR devices.
3. The OSVR community – a global collection of professionals, academics and VR enthusiasts and use and contribute to OSVR



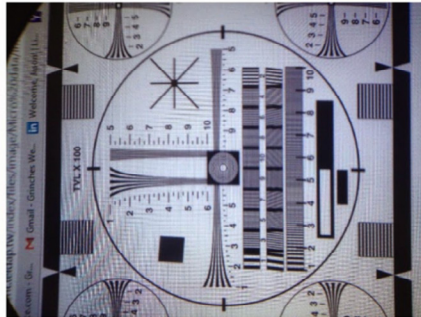
This image shows the breakdown of the OSVR HDK. You can see:

1. Two high quality, dual asphere eyepieces
2. 1080p display
3. Control electronics with embedded 9-sensor motion tracker
4. Ergonomic packaging with several hacking features

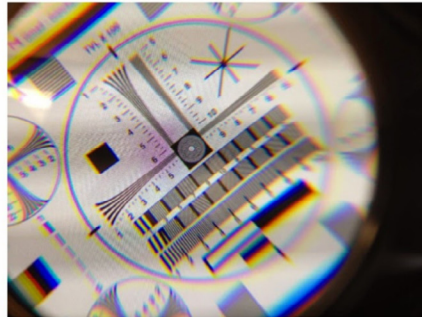
OSVR HDK Optics

- Very low geometric and color distortion
- Sharp image throughout

OSVR HDK: Dual-Element Optics



Typical Single-Element Design



4

This image shows the performance of the HDK optics.

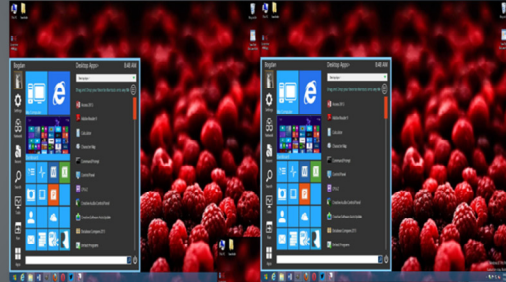
These photos were taken a few months ago using a test pattern which was displayed on a 5.5" display, much like the one inside the HDK. We then used an iPhone 4 camera to take both photos. The left photo shows the test image through the HDK dual-element optics. The right photo shows the same test image through a popular single-element eyepiece.

You can see a stark difference in the clarity throughout the image as well as in the HDK optics' ability to control both color and geometric distortion

For more information about the optics, see the blog post "How things work: the dual-element optics of the OSVR HDK" at:
<http://vrguy.blogspot.com/2015/01/how-things-work-dual-element-optics-of.html>

OSVR HDK: Additional Unique Features

- Open source
- On-board video processing
 - Real-time image rotation
 - Convert full desktop to side-by-side
- On-board USB 3.0 for expansion
- Compatible with wireless video links
- Removable cover
 - Positional tracking
 - Leap Motion



5

Other than being open-source and having excellent optics, the HDK has other unique features:

- The ability to accept both 1080x1920 and 1920x1080 video signals. When sensing 1920x1080, the HDK performs real-time image rotation. This is a useful feature when seeking to mirror the output of the screen, or when seeking to operate the HDK with a wireless video link.
- USB 3.0 ports that allow connecting to other peripherals
- Removable front plate (CAD files available in the open source project) that allow installing cameras or other devices. For instance, Leap Motion is not preparing an OSVR faceplate that includes an integrated Leap Motion sensor

What Is the OSVR™ API?

The OSVR™ API is a multi-platform, standardized interface to virtual reality devices and peripherals.

Designed to encourage adoption of and innovation in AR/VR.

With the OSVR API, developers can:

- Discover, configure and operate a wide range of devices through standardized, abstract interfaces
- Mix-and-match devices (e.g. hand trackers) with analysis plugins (e.g. gesture engines)
- Write once, run on many hardware/platform configurations

6

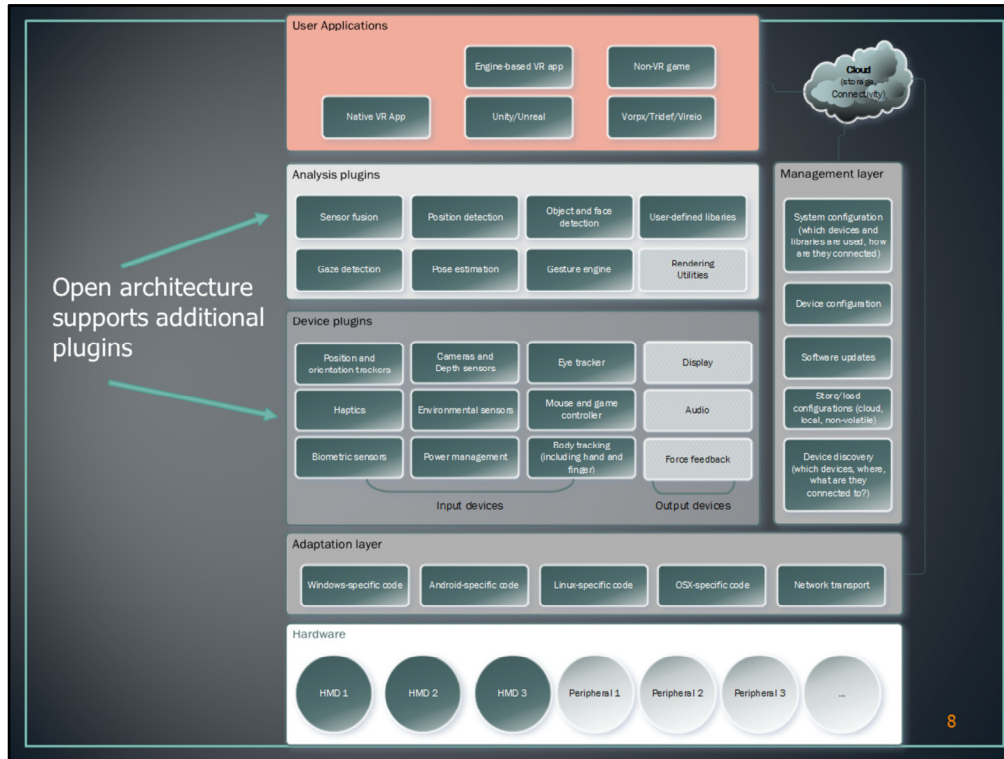
The OSVR software framework is independent of the OSVR HDK. They work well together, of course, but the software can be used with other HMDs (such as with the Oculus DK2) and the HDK can be used with other software frameworks.

OSVR API License

- Software available under Apache Software License, version 2.0
 - <http://osvr.github.io>
 - Can change and redistribute
 - Can use proprietary plugins
- Choice of license intended to encourage participation of both academic institutions and commercial firms

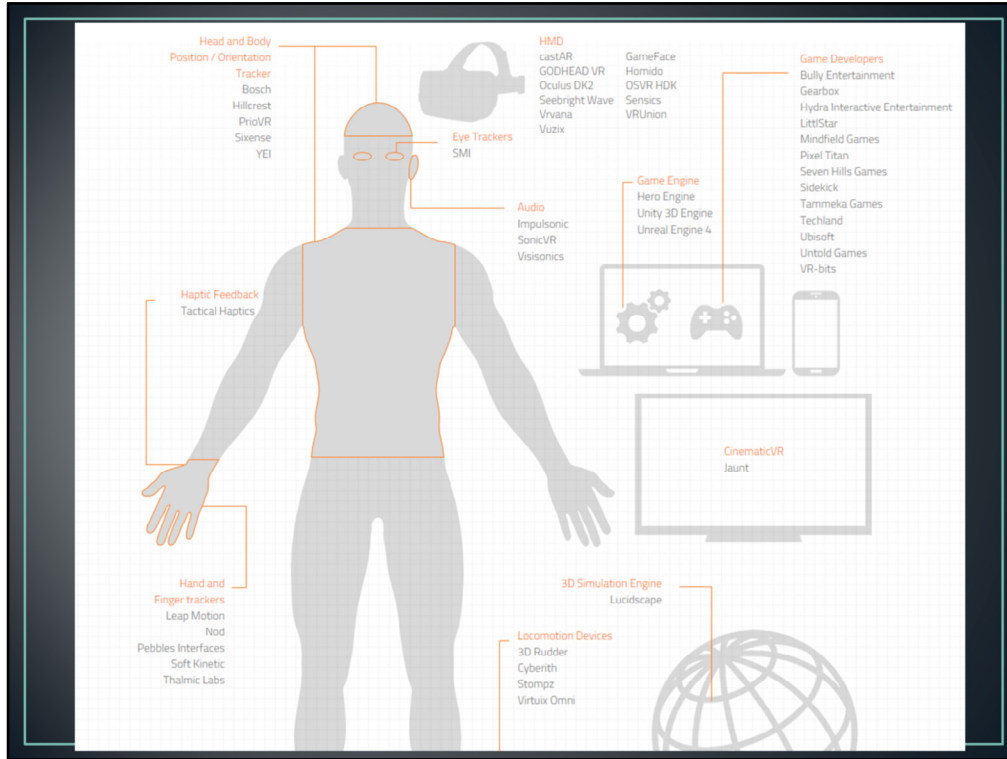
7

OSVR software framework is open source and free. It is licensed under the Apache 2.0 license which means it can be used for any purpose. The license does not require to contribute changes back to the community, though of course we would love to accept all contributions.



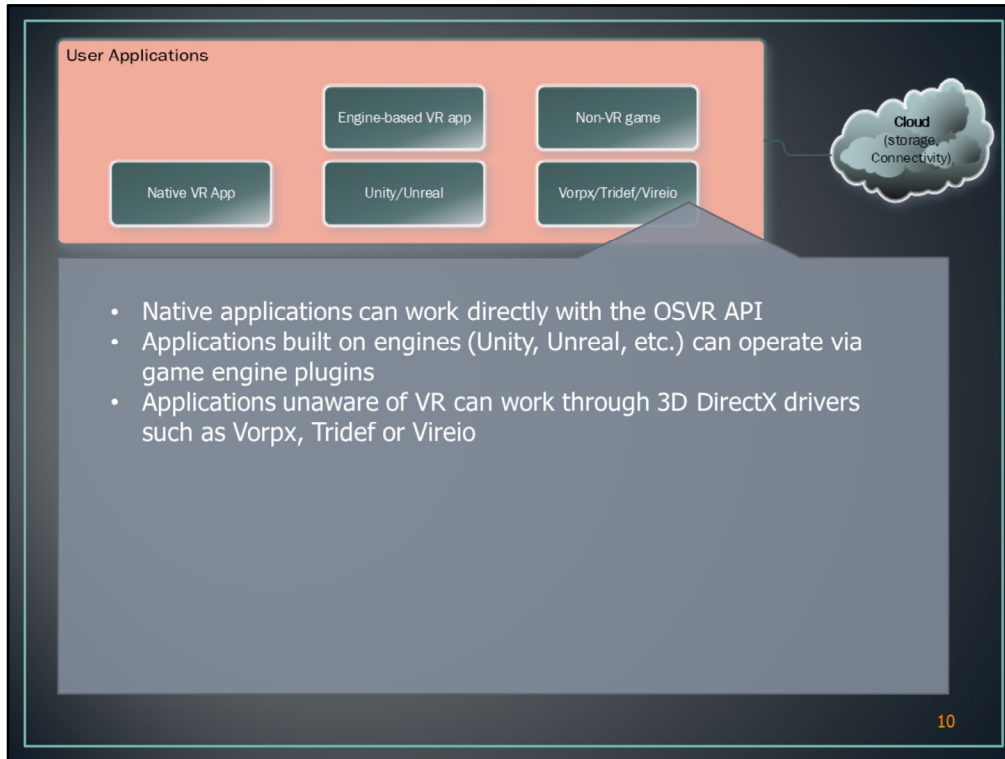
This slide shows a high-level diagram of the OSVR software. It provides a layered approach:

- **Hardware layer** accepts a wide variety of HMDs and VR peripherals
- **Adaptation layer** provides OS-specific adaptations (for instance, working with a USB port requires different libraries in different operating systems)
- The **device plugin layer** provides a standardized interface for various kind of devices
- The **analysis plugin layer** turns data from the device layer into high-level information
- The **management layer** stores and loads system and user-specific configuration locally and in the cloud. It also provides services to download device plugins on-demand, detect when software updates are available and other utility functions.
- The **application layer** allows applications to access the OSVR framework. This can be done through direct access or through a plugin that connects to popular game engines such as Unity or Unreal.



Since introducing OSVR in Jan 2015, over 90 partners have joined OSVR. They include a wide variety of companies that produce HMDs, peripherals, games and more. Check out the latest list at www.osvr.com

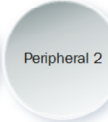
OSVR welcomes new partners every day.

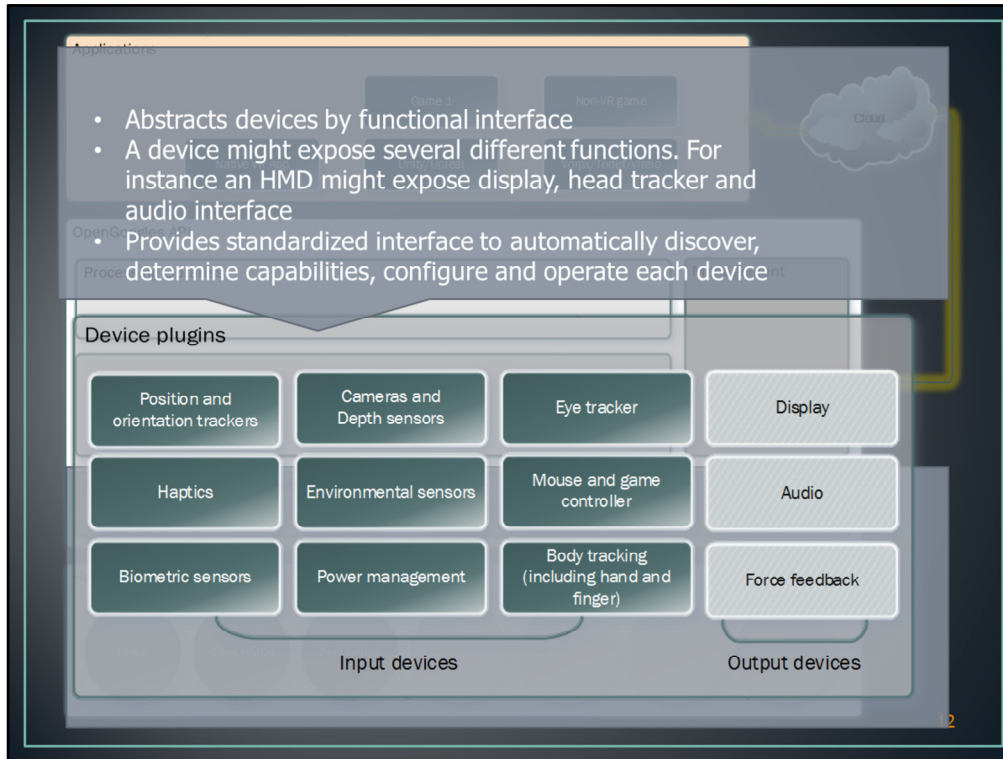


Applications written on game engines can interact with OSVR via dedicated game engine plugins. Alternatively, an application can directly access the OSVR "ClientKit" interface.

- Supports a wide range of displays (HMDs and other display types)
- Supports multiple classes of devices such as:
 - Motion trackers
 - Eye trackers
 - Position trackers
 - Depth cameras
 - Haptic controllers
 - Biometric sensors
- Discovers compatible devices

Hardware





Device plugins connect to physical devices and expose interfaces – pipes of data – to higher layers. For instance, a plugin for a VR goggle can expose a display interface as well as an orientation tracker interface corresponding to an embedded head-tracker. Many device types and dozens of devices are supported.

Analysis plugins

Sensor fusion	Position detection	Object and face detection	User-defined libraries
Gaze detection	Pose estimation	Gesture engine	Rendering Utilities

Analysis plugins turn raw data into useful information:

- Video image → identified face
- Hand position → recognized gesture
- Input from multiple position sensors → better data

13

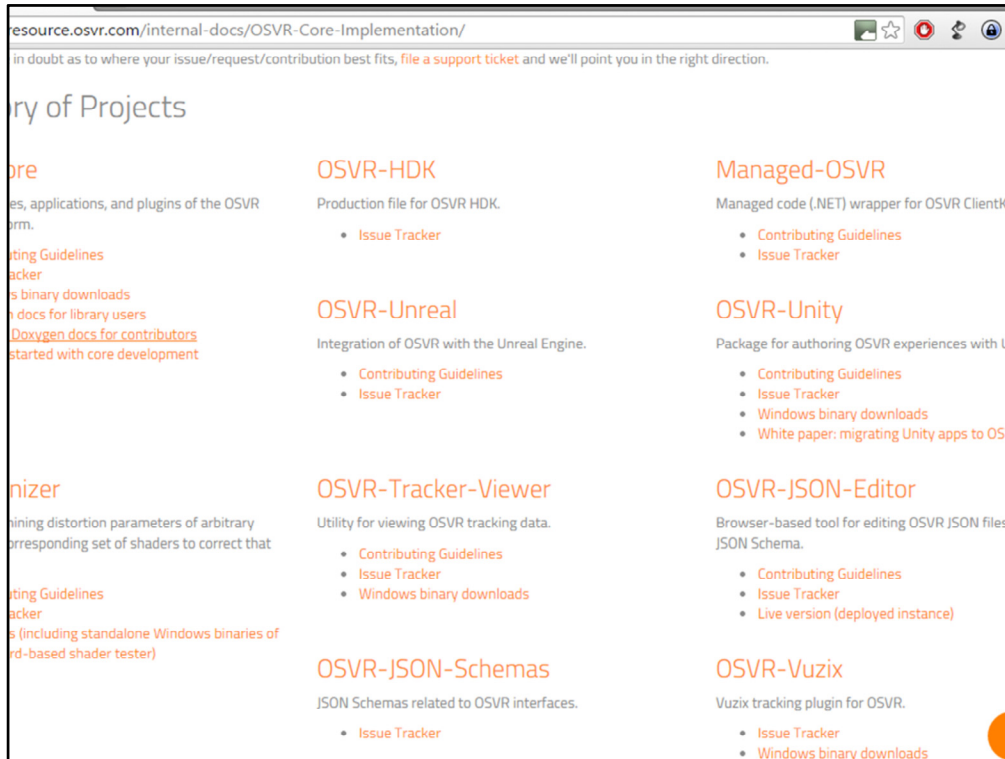
Analysis plugins are software modules that convert data from lower-level device plugins into higher-level information. For instance, a gesture engine plugin can convert a stream of XYZ coordinates to a recognized gesture.

Examples of Analysis Plugins

- Pose estimation
- Face detection
- Distortion correction
- Object/marker detection
- Gesture engine
- Position detection
- Gaze detection
- Rendering
- Directional sound
- Speech recognition
- Redirected walking
- Store to log file
- **Additional libraries can be added**

Libraries are automatically discovered

14



The OSVR source code is available for use, download and modifications at osvr.github.io

Check out our various projects including OSVR core, Unity plugin, Distortion measurement tool, the HDK source code, latency tester and many more

For Additional Information

- OSVR Resources:
 - <http://www.osvr.com>
 - <http://osvr.github.io>
- Sensics (founding contributor to OSVR):
<http://www.sensics.com>
- Yuval Boger's blog: www.vrguy.net