VRIF - OpenXR

Paul Pedriana

The OpenXR Working Group

What does XR mean?

- In a nutshell, extensible reality

- VR / AR / MR

- OpenVR was already in use, but OpenXR has a wider target than VR anyway

- We are working with Google Tango and Microsoft Hololens teams, for example

VR, AR, and MR?

- **OpenXR is primarily targeting VR**

- **But VR a lot of overlap with AR**
  - Flexible viewport definitions being considered for OpenXR serve both VR and AR, for example

High level interaction

- **Applications and graphics engines use the same application interface**
- **VR devices use the same device layer**
- **Runtimes from various vendors implement the application interface and the device layer interface**
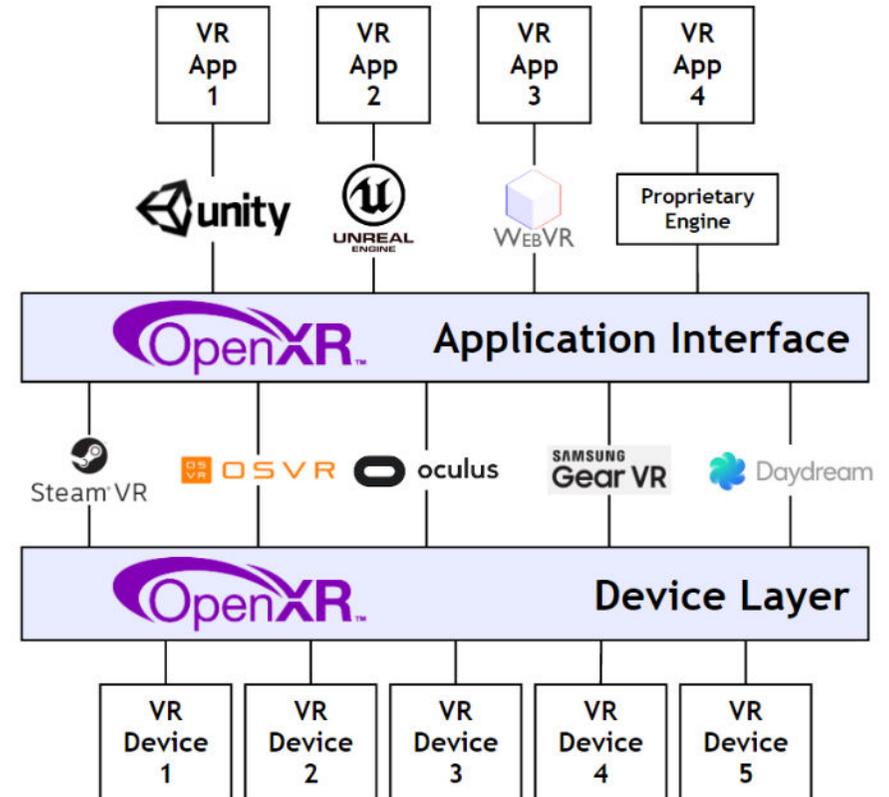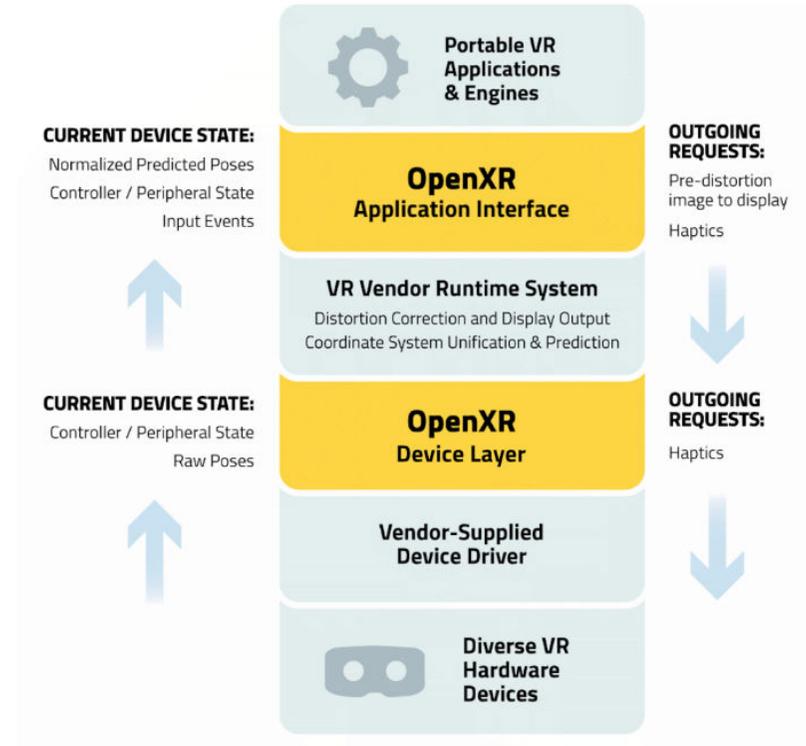
Application Interface

- **Enables VR applications**
- **Provides:**
  - VR drawing interface
  - physical tracking interface
  - device input interface
- **Supports any modern platform, including both mobile and desktop platforms**
- **Supports multiple graphics APIs**
  - OpenGL, Vulkan, D3D11, D3D12, Metal
- **API design resembles Vulkan**

Device Plugin Interface

- **Enables VR hardware to portably work on a variety of runtimes**
- **Provides VR hardware configuration flexibility**
- **Provides room for VR hardware innovation**

API Flow

- **Flow is bi-directional**
- **Applications call through the runtime and possibly through the device layer**
- **Devices call through the runtime and state is reflected through to applications**

CURRENT DEVICE STATE:
Normalized Predicted Poses
Controller / Peripheral State
Input Events

Portable VR Applications & Engines

OpenXR Application Interface

OUTGOING REQUESTS:
Pre-distortion image to display
Haptics

VR Vendor Runtime System
Distortion Correction and Display Output
Coordinate System Unification & Prediction

CURRENT DEVICE STATE:
Controller / Peripheral State
Raw Poses

OpenXR Device Layer

OUTGOING REQUESTS:
Haptics

Vendor-Supplied Device Driver

Diverse VR Hardware Devices

External Collaboration opportunities

- **Standardized display specifications**
  - e.g. EDID extensions for VR displays
- **Standardized VR hardware systems**
  - e.g. reference device layer implementation
- **Graphics API (e.g. Vulkan) extensions for VR/AR**
  - e.g. support for exposing display timing synchronization
- **Graphics hardware support for VR/AR**
  - e.g. support for variable density (non-rectilinear) rendering
- **OS support for VR/AR**
  - e.g. support for private displays
- **Web browser integration**
  - e.g. support for OpenXR in web browsers
- **Extended display functionality**
  - e.g. anti-ghosting features such as pixel overdrive
- **Innovative hardware**
  - e.g. advanced eye tracking

Questions?