

Sean White, keynote.

Reminiscences from the Placeholder Project (1993). It was immersive but very expensive.
VRML 1994. A-Frame, 2016.

- The Long Now: how do you enable VR experiences that are unique to the Web?
- Account for the range of future mixed reality experiences?
- Use VR to make life better in the real world?
- Move Fast!

WebVR Intro

implementation status, obstacles, future plans

Megan: we'll start with an intro from Brandon on WebVR status and then go into implementation status from implementers

Brandon Jones, Google

This is a quick intro on what WebVR is and where the different browsers are at.

A brief history: this whole WebVR thing literally started in a Tweet exchange, with Vlad suggesting how to plug a browser into the first Oculus kit. It seems so consensually Web to me: no funding, no managerial discussion - just two guys doing it in their free time. And look where we're at - that's the kind of innovation the Web enables.

The initial version of the spec was very different from what we see today - it was based on the hardware available then, built around DK1 (DK2 was not available yet).

The API has been changing a lot, with lots of cooperation between vendors. It's getting better and better. What we've called 1.0 has been revised to get up to par with the latest hardware.

Microsoft has jumped in to contribute to the spec, Oculus announced WebVR support.

The state of the art is what we call WebVR 1.1 - which is likely the last version to be numbered - it will just be WebVR after that. This new version is based on great feedback from Microsoft to make sure WebVR can work with their platform, removing guess work for the developers and tightening up behaviors.

There are more non-backwards compatible changes that are upcoming - but we have to do them now. We're getting feedback from lots of people doing VR, feedback to make sure we have the right set of capabilities for the long term.

We are working on bringing Web Worker compatibility which is quite important - the main thread in JavaScript is a big mess, and workers allow to remove some of that constraint, even to the cost of backwards compat. We're also getting feedback from a Web-platform perspective, making sure we get the right model.

What's next? At a mundane level, we want to keep extending the API - VR layers; we want to allow more WebGL extensions targeted for VR (e.g. multiviews). WebGL 2 is going to come out reasonably around the same time of WebVR. And likewise, we want to get ready for WebGL next.

We also need to keep track of evolutions in the VR hardware.

We also need to focus on tooling - there were great announcements from Oculus recently.

Some people were asking about this as being a competition to what I've been working on - but it's a non-issue, it's actually great - nobody is trying to get control or take over the ecosystem.

We need more tools, more browsers, more support for people who want to build content. There is no bad side to doing that.

We have decades experience in the browser which we have been ignoring - we need to bring Web & VR to get closer.

We need to look into support new media formats that are VR native - gltf (not specific to VR, but better support in browsers would be fantastic). New media formats for video and audio that are very spatially aware with lots of info on the environment - we don't want every single developers to reinvent how to get that on the screen.

We need to make sure audio is a first class citizen in VR - there is lots of interesting innovation on the audio side. Good VR without audio is missing half the point - it should not be an afterthought.

We want to make sure the Web enables AR scenarios. [screenshot of hololens user]. WE want to enable full use of capabilities of Hololens or Tango capabilities.

In the end, we really need to think about how to move past this of the current VR experience of a modal app - VR is a very exclusive thing where you launch an app, you use it for a while, and then you move to something else. If we want VR to be a general computing platform, we need to move beyond that. It's extremely important, especially for the Web.

Most people here would agree we're not where we want to be for VR yet.

How do enable composition of activities, allow overlap and still keep it functional. This will require lots of experimentations and failures, but we will get there.

Browser updates

Megan Lindsay - update on Chrome. We are building support for WebVR 1.1 in Chrome for Android, including the game extensions. We're targeting M56 for [daydream](#), should be in beta in December, and stable release in January. We will first release as an origin trial - web sites need to request a (free) token to get access to the API. [Origin trials](#):

The API is still changing quite a bit, so this enables us to get feedback from developers before a stable release.

There will also be cardboard support coming soon.

We're also working on WebVR in Chrome for Windows desktop, with plans to support both Rift and Vive - not sure if we're targeting M56 or 57, as origin trial.

Beyond WebVR, we're working on a version of Chrome that supports browsing in VR for regular 2D sites. Timeline towards first half of 2017 for daydream, with later support for desktop.

We're also interested in standardization on 360° videos, and we also want to look beyond WebGL with declarative VR. We want to enable existing web sites to add VR features without a complete rewrite.

We also want to look at AR but we're not sure what this would entail yet.

Frank Olivier, Microsoft

WebVR is in development for Microsoft Edge, it will be coming up in an insider release at some point. It's built on Windows holographic APIs. We are also very interested in advancing the specifications - we think there will be needs in this space for years to come.

If you maintain WebVR content, we're definitely interested in running it in our browser, to test it beyond our test suite.

Justin Rogers, Oculus

Carmel Developer Preview based on chromium m55 should come pretty soon, with support for WebVR 1.1, and limited support for Gamepad. Target Gear VR and Touch controller. No 3D motion integration.

In Nov 2016, developer release of Carmel, and of React VR.

We'll do another update with M56 in Jan/Feb 2017

We will work with other partners also work in other browsers.

Chris Van Wiemeersch, Mozilla

We started in 2014 as an experimental project based on Brandon and Vlad's work, who is also involved in WebGL. We had an API and wanted to see if we could render content in a browser. The workflow is: you go to a Web site, you click on a VR button and you switch to a stereoscopic view.

This was a demonstration that this was possible.

It wasn't until Justin Rogers did a hackathon with a deep dive on the PoC API, and made lots of great suggestions to improve performance, developers ergonomics and API.

<http://www.justrog.com/search/label/WebVR>

We started a [WebVR Community Group](#) - it has 100+ participants, it's free, chaired by Brandon and myself.

As we introduced more hardware support, we had to update a number of our assumption in the APIs.

WebVR is getting a lot of legitimacy with "native" VR developers.

Casey and @@@ worked on a prototype of what a VR browser would look like, demonstrating e.g. how to click on a link, transitioning to a new view.

This is built around iframes - we don't have hooks for handling navigation transitions.

We created a backwards-compatible VR browser to find deficiencies in the WebVR API.

In the demo, we navigate in regular 2D pages, controlled by Xbox; when moving to a 3D page, it switches to immersive 3D view. This made us realize of some deficiencies in the tooling - that's where a-frame came in: it catalyzes the creation of content and also helped us assess the API is viable.

[these examples show VR doesn't have to be gaming or action oriented - it can be meditative]

[this demo illustrates the performance and interactivity you can get in Web browsers for high end VR]

<https://iswebvrready.org/> shows interested browsers, support for the various APIs - it's collaboratively maintained, including by other browser vendors.

Casey Yee, Mozilla

It's crazy to see all the work that has been happening for past couple of years. The platform work is headed up by Kip @@@ who couldn't make it today - I'm speaking on his behalf.

A lot of the focus has been around getting the API implemented, making it work in our browser. The APIs are available in Nightly, with support for HTC5, Oculus Rift, and working on OSVR(?) support.

As we continue with implementation, we are trying to get a pipeline and process to bring these APIs in production form. We have 1.0 API support, and we're planning to have these APIs available in the stable browser for Firefox 51, available in Q12017.

It will be available in a production browser, so available to lots of developers - which is very exciting.

It's going through our beta API program.

Beside performances, we have been working with other teams needed for VR: audio, gamepad, WebRTC for co-presence - lots of exciting things coming up.

Laszlo Gombos, Samsung

I'm going to talk mostly about mobile VR - as enabled by Gear VR.

The Gear VR has an external touch pad, and an external usb port which enables to plug more input devices.

Gear VR has pretty large field of view, it has improved tracking - when docked, it doesn't use the mobile device sensors but its own IMU.

We built a VR-first browser - it not only supports WebVR, but it also enables browsing the regular Web in VR. For instance, to consume existing medias in VR - kind of the TV use case.

This is good way to get people hooked up on VR content.

We try to take advantage of the space around you for interactions.

This shipped in Dec 2015, 10 months ago. We got pretty positive feedback.

First stable release in March 2016, with WebVR support based on M44 Chromium. The WebVR support has to be enabled specifically though.

We are 1.0 since April, and our latest release was in August.

This is a VR-only product that users have to download - we reached over 1M downloads!

If you are in mobile and in the samsung browser, watching content/video, and dock the mobile in the headset, we will switch you seamlessly to the VR browser.

We support voice input for text. We have curated content.

We have an extension of HTML video support.

We allow the Web page itself to change the environment (e.g. show a background scene image around the browser window in immersive view) - this is already shipping, the skybox JavaScript API. "Progressive enhancement" from 2d Web.

[showing demo of Skybox API]

What's next: we want to enable WebVR by default, which requires improving performance; we want to bring more control to the VR environment e.g. with the Skybox API. Discussing what bringing the latest improvements of Web Platform to VR is also quite exciting to me (e.g. Progressive Web Apps).

VR user interactions in browsers

Designing the browser (Josh Carpenter, Google)

Josh - been working on figuring out how we want the experience of surfing the web with a VR headset. As designer, want to create possibilities - create the conditions so others can make experiences.

By the way - that (demoed on screen) is all CSS.

“Cambrian explosion” of browsers. For years we’ve been trying to cram browsers in those tiny screens. In comparison - VR is an infinite canvas! We are about to see an explosion in browsers.

What are the things we think all browsers should do consistently? Somewhere between pure freedom and a few guard rails:

- Avoid dead zones - give developers freedom to design
- Facilitate speed - this is where the web can compete (not necessarily graphics but speed, especially speed of loading). Show demos of transitions.
- Big paws and oil tankers - obligation to learn from nimble VR-first projects

Link traversal in WebVR (Chris Van Wiemeersch, Casey Yee, Mozilla)

Casey and Chris. Colleagues at Mozilla. Always wanted to move between content in VR. Hyperlink is a fundamental portion of what we think the web should be, want to port that to Web VR.

Web evolution. Move from Page to World.

Our responsibility, what we have to do to preserve future of the web as we know it:

- User Control - choice of where to go and how to get there
- Security - no surprises. Protect the user from bad surprises, cross scripting etc
- Openness - enforces interop. Make sure these aspects of navigation remain

Demo (maybe??). OK, no demo. Videos.

- Representation of the link as some kind of orb in space

Note this is not something discussed in depth in the webVR community. This is an invitation to come and discuss the how - this was only the why.

Hand tracking and gesture for WebVR (Ningxin Hu, Intel)

Slides: <https://huningxin.github.io/webvr-hand/#/>

Want to talk about hand tracking and interaction.

Quote Brandon Jones: *"They watch some demos or play a game and walk away saying how impressive it is, but almost everyone makes a remark about how they wish they had hands."*

Today's techniques generally using depth camera. Create depth map.

Two kinds of hand tracking: skeleton tracking and cursor tracking

Demo of skeleton hand tracking - for avatar hand tracking. More computing intensive, thus challenging for mobile devices as it impacts the battery life and may introduce latency. No standards for this today, typically using websockets. Area for standards?

Cursor tracking models the hand as one pointer in space. Lighter weight, requires less computing, and can be more stable and accurate which makes it better for accurate UI control.

Another aspect is gesture recognition. This delivers high level events. Recent trend to use hardware acceleration to handle this. This area is quite heavily patented, could be problematic.

Quote *"So my prediction is that in five years we'll see good avatar hand tracking and gesture-based simple interface control"* - Michael Abrash

We need to get open web ready for that.

Coordinate Systems, Spatial Perception in Windows Holographic (Nell Waliczek, Microsoft)

Nell - one of the developers on the Windows Holographic dev platform. Now enabled in mainstream PCs.

Want to talk about spatial perception.

Need to know where the user is. Hololens does that without external HW. But we also need persistence of the world around us. "Go anywhere" with hololens (photo in ISS).

What's the catch? No such thing as fixed, global coordinate system. Landmarks may not be where they're expected to be. Other issues such as low lighting, obstruction etc.

Solution? Spatial perception.

- Attached frame of reference
- Stationary frame of reference
- Spatial Anchor

Best practices (lots of bullets!) @@ need link to slide

Ray Input: default WebVR interaction (Boris Smus, Google)

<https://docs.google.com/presentation/d/1k1dAEKB7axlOum5OiwNTcq4vVpHjSlbQdkjeEDdVKgw/edit#slide=id.p>

Want to talk about Ray Input. But first, go back to 1998. The web had Pages, scrollbars, blue links etc. Imagine if we only had a mouse and no interaction standards to start with?

2016 VR world? Yeah, that. What should the input patterns be? Currently a lot of experimentation, which is both good and frustrating for end users.

Proposing laser-pointer style defaults, and fallbacks for non-VR platforms.

[Video] shows all the interaction modes, depending on platforms:

- Mouse interaction. Look around with mouse lock, ...
- Touch interaction. Touch panning, tapping
- Cardboard mode. Tap anywhere and interact
- Daydream - ray-based interaction

Fairly simple API, open source.

Questions: where do you position the daydream controller. Arm model? Give it orientation + position. Built open source simulator. Feedback welcome on github. @@ link?

Samsung VR Browser for GearVR learnings (Laszlo Gombos, Samsung)

Laszlo continues from earlier talk. Want to talk about how to solve issue of tab management in VR. Currently feedback is along the lines of “too complicated”, or “have to move head too much”.

Came up with another model, but then people started asking for more windows.

Mike Blix takes the floor to talk about Skybox API. Goal: easy enhancement for VR. Change skybox environment in VR web browsing environment. Discussions on how to declare.

API is fairly simple. Use case for preview of webVR content, for example.

Group Discussion / Joint Summary

Opening the floor. Potch reads questions from slack.

1. How do you bridge the gap [didn't get details]

Nell - tell me more about what you'd like to see?

2. Question on raycasting. How do you have arm model without extra input?

Boris: heuristics based on fixed offset between head and elbow.

3. Also on raycasting. How do we borrow more interactions from 3D game developers. Look to huge library of 3d games and borrow beyond interaction with basic pointers.

Boris: still very new

4. Philip S - great start, but there are so many other interaction patterns we want to bring in and make reusable. Maybe topic for breakout tomorrow.

(Brandon - there's a slack channel we just created)

5. Jim Bricker - question on gesture control: anyone thought of using device everyone already knows (smartphone)

Ningxin: depends on smartphone vendors adding the right sensors.

6 Potch: how do we find basic standard set of gesture given cultural variety

Ningxin: [missed response]

Philip S: interesting thing we've done: hand facing out interacts with the world, looking at palm of your hand shows menu

7. Sean W: worth thinking on IEEE VR, not to copy them but worth adopting some of their work on e.g. click

8. Question for Samsung folks on skybox: can you do things like gradients.

Mike: you can generate color image dynamically, etc. Sharing details on slack.

9. Chris vW: how do you imagine implementation of [??] (skybox, scribe assumes?)

Laszlo: you have basically HTTP headers, meta tags or js. Discussion on which one you want. Our intent would be to standardise.

10: Nat Brown (Valve): not enough thinking about room scale. Transition important in larger areas.

Potch - leads us to goals of this session - identifying need for basic primitives. Maybe better primitives for location and room scale.

Brandon: reiterates needs for common primitives, using the idea of scrollbar as example. This is like native apps where they all control differently. There is value to that, to having things purpose built. But we want to give the web primitives that work in more predictable way.

Philip S: we also should think about AR at the same time, because it is different - there are objects in the world you want to interact with. Need to think about that.

LucC: unifying by thinking of AR as different only in transparency?

Brandon: main difference is environmental knowledge.

Nell: if you're not trying to bound user to where they are in environment, none of this is hard. Transparency is only a piece of that. Simultaneous connection to reality of place too.

Chris vW: webVR is the more mature of the immersive APIs. But we're going towards 3D browsers, while developers are used to 2D. It's really about getting the web into 3rd dimension, not just about VR, AR, MR. 3D web.

Philip S: want to add to the AR discussion. In a VR world everything is in that world. Different interactive distributed simulated environment.

Diego: 1 to 1 mapping of space to movement for room scale environments. Ability to integrate real objects into the VR world. We can explore this.

Dom: tries to summarise things needing attention from standardisation perspective:

- Gesture recognition (but culture)
- Perception API
- User input model
- Skybox API
- Keep AR in mind for interaction model
- transitions / flash of light
 - preload next page/experience (header?)
- flow / simple layout (via library?) for 3D
- collision detection / event management
- priority zone (avoid getting lost)
 - how to navigate VR experiences / design navigation mechanisms
 - guiding the user: what is the standard model to get oriented

Jason Marsh: (pre)loading/whitespace transitions. How do we know how that feels/should work?

Chris vW: ongoing work on transitions. Going to help 2D but will also help VR immensely. (@@link?)

James (??): simple layout, flow. Like in CSS. Also need something for collision detection.

Jeff Rosenberger: need for a priority zone so user does not get lost. I.e., a place where new apps appear "in front"

Casey: navigating VR environments. Need to design wayfinding mechanisms.

Poch: good segue into accessibility set of lightning talks.

Accessibility of VR experiences

Accessibility Not An After Thought (Charles LaPierre, Benetech)

<https://benetech.app.box.com/s/0vgnivm0pawfti0izzs11srppp9dfn1o>

Overview of accessibility at W3C: crosses a number of domains and disabilities. Disabilities to consider include auditory, cognitive, neurological, physical, speech, visual.

Needs to be part of all aspects.

Multimodal: Accessibility is inherently providing information from modality in another.

Both inputs and outputs need to be accessible.

Example, an accessible shopping experience. If one user uses gloves to interact with video, another visually impaired user might have self-voicing option. Proximity. More detail. Speech input,

Consider multiple disabilities, e.g. deaf-blind. May need another input or one that hasn't yet been invented.

Consider all the inputs in a shopping experience. Sight, touch, description; how would we present the "shop for a dress" experience to all the senses, including for someone who lacks some of those senses.

Make accessibility a first-class citizen. Work with accessibility groups.

Built accessibility use cases, look for ways general interfaces support accessibility.

VR to help the disabled, e.g. to help train with prosthetics...

Consider lessons learned from making videos and images accessible.

Browser UX in VR (Justin Rogers, Oculus VR)

https://onedrive.live.com/view.aspx?resid=DB996F916EA4CF47!60653&ithint=file%2cpptx&app=PowerPoint&authkey=!AAsddctAo_Xqgsc

I'm talking about making Web content accessible to all. Is 2d web accessible in VR? The web still has lots of 2d content. Input is limited.

What are the most used applications? Virtual desktops and browser VR. They're sort of uncomfortable: it's an accessibility problem for all of us.

4 key areas to look at:

- High quality reprojection.
 - Fix text. Cylinder surfaces.
 - Good UI, positioned for head comfort.
- Nail input.
 - Most inputs can be resolved with gaze

- Big hit-targets, Hit-target attraction
 - Link disambiguation UX
 - Voice commands, simplified ux
 - Limited context
- Design for security and trust
 - anti-spoofing
- Redesign for the medium.

Mixed reality (Rob Manson, awe.media)

<https://docs.google.com/presentation/d/1FHbh0XrOcsPNs5pjJs9oFPU-D5p2bTZ2wVtkGFVz5wk/>

I'm from Awe Media. Pitching development of AR in the web browser, this is a critical time. Mixed reality on the web. Real-world input into virtual space. Geometry of tech shape how we use it, changes innovation. Consider the evolution of the camera, from camera obscura, brownie with viewfinder, SLR, LCD screens, selfies.

Geometry of spatial apps in HTML5. Objects in the real-world that exist beyond "far" How can we bring depth in?

Orientation and position.

Question: what about color-blindness?

Charles LaPierre: We want to be able to allow shifting of color spectrum, manipulated by APIs or by user. That's possible now. We can use VR to demo what a person with color-blindness sees in different scenarios, and how to make it more accessible.

Brandon: I am red-green deficient, so this is important to me. In VR today, there's already some post-processing, so we could add color-shifting. Also, volumetric data.

Shannon: I worked on ARIA spec, declarative attributes on HTML DOM tags for purposes and intents. We should look at how that can be adapted to WebVR.

On-click element, declarative.

LucC: Look at solutions from "real life", diversity of accessibility solutions, we should make it possible to adapt these in VR.

JamesBaicoianu: Re post-processing, look at CSS, users can specify styles that override the web author.

Anssi: Declarative, learn from CSS. How do the existing accessibility tools for the web fit into WebVR? What abstraction layers?

Charles LaPierre: We need low-level APIs that can interact with existing assistive tech. That exists, e.g. braille display, tongue sensors for visualizing imagery. Where APIs exist, you should be able to interact with those in VR. Give developers hooks to those APIs. Personalization is a big part of accessibility: not every blind person can read braille, so let them work through the modes they do use.

Philip: HDR displays will introduce new issues. Extended color gamut makes RGB look dull. Contrast ranges need to be mapped to displays.

Casey: Lots of accessibility already built into the browser. Use it.
There's also machine accessibility. How search and links work.

Potch: I'm looking forward to VR alt-text.

Is there a need in primitives to separate visual from spatial? How do we take the interactive and spatial components and not make assumptions about users' characteristics?

Chris: A-Frame meant to be the JQuery for VR on the Web. document query selector proved the value, then got standardized and implemented.

Big q: Should every entity be represented in the DOM?

Q from slack: klausw: accessibility: Roomscale/6DOF has its own challenges, think about wheelchair-bound users, people with only one arm, very short or tall people. Many of the same accessibility/ergonomics issues as for real-world locations. I remember having to lift up my daughter to reach the virtual pot in Job Simulator.

Brandon: Cool utility @@.

Possibility for extension utilities to help with common tasks, e.g. "make me taller, to reach items". As patterns emerge, maybe they become part of the platform.

Klaus: do we have the same expectations for all sites? Difference between action games and shopping?

Charles: For an image, we can have a tactile representation, but feeling it doesn't make sense without a tour to describe what you're feeling.

3D audio is great for people with visual impairments. You can create amazing 3D audio games; so games can be made a good experience for people with disabilities.

@@: MSR game with no visuals.

Justin: some preferences we've exposed in WebVR, e.g sitting to standing pose. These keys can be valuable to improve the experience. Be careful about privacy considerations. As we tailor experience to a user in a wheelchair, are we exposing to fingerprinting/tracking?

Q regarding text.

Justin; we're looking at SDF fonts, we don't have a canned solution to improving 2d text.

Brian Chirls: top-down vs responsive design. If you take accessibility into account, you get lots of great stuff for free. Even a person who has all the VR equipment for roomscale sometimes wants to sit on a couch. Can we build features you don't have to "turn on"?

Anssi: list of topics. [on screen]

@@: API that faces up, and the API that faces down to the hardware. Accessibility works well when device devs can experiment with serving user needs.

[lunch]

Multi-user VR experiences

Internet-scale shared VR (Philip Rosedale, High Fidelity)

Philip: We're working on an open source license that enables you to do face-to-face interaction. I'll present some assertions and discuss what we're doing.

[Assertions slide]

There are two reasons that this is disruptive. The first is the order of magnitude of interactions, the 2nd one is face-to-face communications. Allowing hands and heads to move naturally is the general use.

Client-server model, enabling interactions, rather than app-store centralized

[New Technology slide]

There is a need for low latency - milliseconds needed

3D audio and avatar motion

Compressed scene description

Real-time scene / object

Distributed servers.

[Areas for Standards and Cooperation slide]

Identity needed - bring your avatar, name and appearance

Content portability - interactive need

Authenticity for assets - not DRM but verify that things are what they say they are

Server discovery - something beyond DNS

From clickable pages to walkable spaces (Luc Courchesne, Society for Arts and Technology)

Luc: There have been many things done on this topic including @@@

[picture of @@@]

Transitions slide

How do we move between experiences

Most people will not care - they will go together without thinking about it

[slides of sharing experiences]

Now with HMV immersion things can be experienced more easily

WebVR slide

This is going to transform the Web into multiple spaces

We understand the need for Avatars to invite people in

The limitations are a problem

Slide of topics

Virtual "situation room"

"On site" design review

Goal is to build virtual teleportation platform

I'm going to show you what we did at SAT over the past few years on that

Video of test work

In 2012 we moved to Kinect

We could extract the forward from the background

You feel the presence of someone from their natural body language

2013 The Drawing Room

By using 3 cameras we could port the system in real time

What we're working on now is ### experiments

We need help - find us if interested.

Multimedia & multi-user VR (Simon Gunkel, TNO)

Simon: I work for TNO in the Netherlands. I'm glad to be here and am very impressed that many of the topics we're struggling are being addressed.

We are looking for attached experiences vs. detached experiences.

At TNO we do applied research so today I want to talk about my first experience and some of the problems along the way.

Say you're sitting on the couch and want to have an experience.

This is a mock up as it doesn't show a headset, etc.

This shows people looking at each other and feeling their presence.

People slide

How to engage people while letting them interact in 360/3d

How to position people

Interaction with environment

Multimedia Objects slide

Synchronization

Dash streaming and tiling

Adaptation of spatial audio

Browser slide

Different browser support webvr

Different browser support of Spatial audio

Performance of JavaScript and webgl

Conclusion slide

You can make a natural interaction but you have performance concerns

Mixed reality service (Tony Parisi, Wevr, on behalf of Mark Pesce)

Tony: Mark in in Sydney. I'm going to talk to you about his new service.

About 22.5 years ago we pulled together enough money to send Mark to the first WWW conference in Geneva

It built the spinning banana running on a server in CERN

We've kind of gone full circle. Pokemon Go delights millions but causes odd behavior

Mixed reality service provides this metalayer layer, binding the real and virtual worlds

Very simple protocol showing 'add' operation

Delete is the reverse of that

And the search is going to let you find a service set of URI's

[demo]

We think the world needs this type of markup.

Copresence in WebVR (Boris Smus, Google)

https://docs.google.com/presentation/d/1k1dAEKB7axlOum5OiwNTcq4vVpHjSlbQdkjeEDdVKgw/edit#slide=id.g184a962562_0_71 Co-presence in WebVR

Boris: As we all agreed we are building an isolating technology

List of components used for demo

You enter and have an avatar you control

You can see other people

You can shrink or grow and sound scales accordingly

3DOF Pose Audio Stream slide

Shows p2p connection

$O(n^2)$ will not scale

[slide showing streams]

Some fun features

Mouth moves

\$\$\$\$\$

This all doable, but it's just a demo

There is a lot to be figured out

How do we do this, what is the path to identity, avatars and payments

Discussion:

%%%%: A couple of the talks spoke of users and finding them in a space. This requires things working together. Has anyone started working on this standardization? Feels like everyone's working on their own implementations.

Tony: We're addressing some of that and it seems like a service layer that's waiting to happen.

Potch: How do you position people in space where Avatars aren't near each other. There is balancing of the needs from many perspectives.

#####: When we're talking about social interactions we're talking smaller numbers. We have differences in real world spaces. It doesn't make sense to have everyone in the world, just the 2 - 3 of your friends that want to talk.

Potch: A more practical question. Where do we see the authoritative database being hosted.

Tony: Good question, who's the authority over a certain space. We hope as the problems become evident we work on them together to solve them.

^^^: There are a lot of projects to make data more distributed and less centralized. We'll have to see how this evolves.

Nicoli: There was a session on latency. What's the real need, is it 100 ms or less? If someone speaks to me and I don't hear it, was it too long?

@@@: I don't know if it's any easier in games, most have a minimal latency, I think we have a lot of work to do on this. We should look at game engines to learn about low latency and figure out how to scale that.

Potch: question about having a private UI and a public avatar. Is there a way to have me see things you can't. It gets to questions of privacy.

^^^: I think people are going to go towards everyone seeing the same thing. I think several solutions will be as literal as they can be.

Wendy: That conversation raises the question to me of how much do we expect the environments to be end-user customizable. How do I bring my own private annotations, accessibility tools, maybe I have things I want to share with friends.

Brad: If you want to record someone's phone conversation you have to tell them. So if you were to bring something into a VR environment and record someone else's VR would that be legal?

%%%: If I try to do something in AR do I need to worry about copyright or do I just get pixels.

Dom puts up list of topics that were discussed in the session.

Discussion on the use of avatars and the security around them

/missed some of this due to hearing issue and that my laptop stopped working for a few minutes/

Privacy/anonymity: for some users and situations, it's dangerous to be identified, for others, it's unwanted. VR needs to accommodate multiple identities and pseudonyms/anonymity.

The Internet is composable. It's not uniform. Will VR be a metaverse or an Internet?

Charles: drawing on that, if you are paralyzed you may not want to show that in VR. You may want to have an alternative presentation.

#####: People react to your avatar and in VR we need to see if people are more respectful to each other.

@@@@: A lot of stuff being discussed here was covered in Virtual World discussions so it will be interesting to see how that's changed over time.

Potch: To what extent is the user agent going to impose restrictions on the character model.

%%%: The idea of smooth degradation of avatars needs to be considered as well.

Authoring VR experiences on the Web

HTML & CSS, (Josh Carpenter, Google)

(Demo of 2D website viewed in 3D)

WebGL - steep learning curve, no consistency of user experience. "Recall the Web of the 90s where we each had to implement our own scrollbars." Z-depth is the drop-shadow of VR.

Permissions model for depth, akin to full-screen, permission to come closer.

WebVR with Three.js (Ricardo Cabello, Three.js)

JS 3D rendering library, open source

threejs.org

(Chrome) Browser extension: WebVR emulator, interact with 3D environment without HMD

(Slides showing JS code)

A-Frame (Kevin Ngo, Mozilla)

aframe.io

Web framework for building VR experiences

Based on three.js

Custom HTML elements: a-scene, a-sphere etc

Demo: Hello Metaverse

Works well with most other JS-based frameworks

Entity-Component-System under the hood

A number of built-in components, easy to expand with custom components

Registry: Curated collection of components

Inspector: Inspect and modify components

A-painter: Paint in VR in the browser

React VR (Amber Roy, Oculus VR)

ReactVR: Bridge needs of web developers and VR developers

React: JS library for building UI for the web (Facebook)

ReactVR, based on three.js, WebGL, WebVR

Key features: Based on React code - diffing and layout engines, code combined w/declarative UI, behavior/rendering in one place, optimized for one-page web apps

(Code example)

Code transpiled to JS

Demo: developer.oculus.com/webvr/ (Hotel Tour)

XML3D (Philipp Slusallek, DFKI)

XML3D: declarative 3D framework extending HTML5

Generic data types + data flow programming (Xflow)

Programmable shaders

Renderer-independent: WebGL + shade.js

XML3D-NG

Uses Web Components, WebVR

Smaller core library, domain-specific components

Shareable web components from remote library

Core data model: Data tables and entries

Code example: Replicate <a-sphere> using attribute binding and core elements

Demo: WebVR plugin: King's Cross Station (did not work)

Webizing VR content (Sangchul Ahn, LetSee)

Some experiences with current VR/AR approaches

Issues: Limited functionality, no way to refer to real world, separated rendering context

Expectations: Mashable, dynamic, responsive to both virtual and real worlds

Demo: A mobile AR web browser

Requirements and opportunities for standardization:

1. Evolution of HTML for VR/AR
2. "Physical things" as resource
3. New media type for VR/AR

AR demos: AR book inspector, MAR.IO AR game prototype, LetseeBeer

gLTF (Tony Parisi, Wevr / Amanda Watson, Oculus VR)

https://docs.google.com/presentation/d/1BRdEGqJFIWk3QOehOxJqM9dIE4kIBNQhIm7UeBaVse0/edit#slide=id.g185e245559_2_28

Real-time 3D asset delivery

No file formats specified in WebGL

Or for 3D in general

.gltf : JSON node hierarchy, materials, cameras

.bin : Geometry,

(and more)

Oculus: A need for a standards format for 3D scenes

Spec and issues discussion on github

"The JPG for 3D" !

Vizor - visual authoring of VR in the browser (Jaakko Manninen, Pixelface)

Publishing platform for VR content on the web (creating, publishing, discovering)

Visual 3D compositing editor

Visual programming language / Node graph for programming

One-click publishing to the web

vizor.io : Discovery system on the web

Discussion:

Q: Relationship between gLTF and a-frame?

A: a-frame intended for hand-coding, gLTF for exporting from tools. A-frame is a JS framework, gLTF a file format.

Samsung research: Encourage browser vendors to support/optimize for higher level formats.
Browser should evolve into high-performance game engines
Also, browsers should natively support glTF

Josh: Would like to reimplement the browser on top of WebGL, then solve backwards compatibility

Q: Already lots of standards, do we need another one (glTF)? (ref. XKCD)
Neil Trevett: Yes, we need one that addresses this use case.

Q: Is glTF positioning itself as the baseline spec for WebGL/WebVR?
A: "Firewall" between WebGL and glTF spec work, no intention of mandating glTF for WebGL.

High Performance VR on the Web

Chrome Android learnings & pitfalls to avoid

https://docs.google.com/presentation/d/e/2PACX-1vRPK_ZAC2mKNwStORRB9vFEzpac3NiDw4zPFjN44wC29FbZyaOF1N4Eyf7_rINqlrBhZYs3AechTkpG/pub?start=false&loop=false&delayms=60000

Justin Rogers, Gear VR Performance Tweaks and Pitfalls

<https://onedrive.live.com/view.aspx?resid=DB996F916EA4CF47!60659&ithint=file%2cpptx&app=PowerPoint&authkey=!ABO1DW0msUPuBec>

"Android & Mobile is really hard"

Building a WebVR Content Pipeline

https://onedrive.live.com/view.aspx?resid=DB996F916EA4CF47!60661&ithint=file%2cpptx&app=PowerPoint&authkey=!AK84_AzJVjRm3qE

WebVR Next with more layers

https://onedrive.live.com/view.aspx?resid=DB996F916EA4CF47!60663&ithint=file%2cpptx&app=PowerPoint&authkey=!AJF8T-vOzNiR_ZE

Posh: reading from Slack - jank caused by GPU uploads - what can we do?

Brandon: no great answer, but for a lot of these things (e.g. texture uploads), there are specific mechanisms that allow to separate out pieces in the texture upload. Right now, you use an image tag to load an image, and once done, you ask to browser to upload it, all of that can be blocking. What you really need is having compressed textures all the time throughout the pipeline, and we should expose that as a primitive to the browser.

@@@. What optimizations are available for 360° videos?

JustinR: lots of big problems for 360° videos - e.g. how to stream it. Can we do a better job of sending information to optimize some of these streaming concepts? The server could optimize which slice to prepare for the viewer.

Paola: how do you measure performance? What are the metrics? What are the tools?

Brandon: in Chrome, we have really cool dev tools, although somewhat undocumented and a bit hard to discover. The timelines give you beautiful graphs with very nice clear ideas of what takes time in your apps - we've some screenshots of that throughout the day.

Klaus: back to the room - to what extent the developers want this exposed to JS? What should it look like? What metrics would be most useful?

@@@JanusVR: is it now possible to do @@@ with renderTexture?

Justin: it's one of the bug we hit in Chromium - I think that has been fixed in drivers

Brandon: in WebGL 1, if you want to use renderToTexture, you can't do multi-sample @@@. Won't be fixed in WebGL1, but WebGL2 will fix this - but you shouldn't deal with that type of approach for VR. For mobile devices, you want to use well-tested techniques from the 90's, multi-path is not there yet.

Justin: for your main rendering process, use a simple WebGL context, especially on mobile

ChrisVW: mirroring can be important for demos esp on Desktop; what's the alternative to preserveBuffer in that context?

Justin: for demos, that's probably OK, although you'll find devices for which that will fail. But don't keep it in production. It's useful for demos or development, even debugging.

ChrisVW: have you investigated asm.js or WASM?

Justin: no

Brandon: mirroring is good on desktop, but never do it on mobile. You can chromecast content from the headset to the TV - this would be the better mechanism, with hardware acceleration. Don't turn preserveBuffer on on mobile.

Anssi: what are the changes needed in Web facing APIs? What needs fixing?

Dom: <https://hillbrad.github.io/sri-addressable-caching/sri-addressable-caching.html>

@@@: WebVR doesn't reflect any performance back to the app - e.g. on texture size, resolution scale.

Brandon: we have performance extensions for WebGL, although there is some additional cost coming from the compositor (which we hope to be able to turn off e.g. on mobile immersive). The concepts behind the requestIdleCallback API of a time budget could usefully be reapplied to requestAnimationFrame

@@@: we need to know how much GPU time you have left, not CPU time

Ricardo: what about controllers? Will they be integrated in the gamepad API or a different one?

Brandon: for one, we want to expose some really basic fundamental interaction models, regardless of what you're working with; on cardboard, the one-button interaction should be surfaced as a touch event, which can then be replicated in more advanced devices.

For gamepad, we should poll all the information at the same time.

Justin: Another crazy idea is to keep GPU cache across navigation, assuming we have the right conditions from a security perspective - this is critical to enable smooth navigation.

Ricardo: I'm wondering if at some point the browsers will give us the glTF model for controllers; right now we have to manually change this.

Brandon: OpenVR has this; I've been wondering if we should do this, not sure how to do it in a Webby way. It would be nice to defer to the browser and just ask for a representation of the controller. But I'm not sure what that API would look like - we could just resurface the OpenVR API, but we probably want something more useful.

Potch: We have ServiceWorker that can do request caching; we could have a way to get the SW to get back the fully decoded ready-to-use asset as a response. Sounds like a logical extension to an existing API.

@@@: catching CPU resources across navigation - we have very limited GPU resources, and have no indication of back pressure from the GPU. WebGL doesn't easily let us recover from exhausted resources. Right now we have to manage this blindly. Regarding dynamic resolution, there is nothing in the WebVR that tells us the target framerate.

Casey: how about non-graphics optimizations: physic, spatialized audio?

Brandon: I would love to see more tooling around audio and making that an easier thing for developers to get their head around. We have great engineers in the Chrome team who do amazing things e.g. the WebVR audio sample demonstrating spatialized audio. I would like to see more tools that handle audio in a more user grokable way, with the best performance outcome. We have the right primitives available, but I find them hard to use.

@@@: MultiView was mentioned as an improvement for WebGL - we have a proposed breakout session for tomorrow. Are there other acceleration extensions we would need for VR?

360° video on the Web

Louay Bassbouss, Fraunhofer, 360° video cloud streaming and HTMLVideoElement extensions

<https://docs.google.com/presentation/d/1-stapAqJr5ICQcXEC7z-PYnTYY1T3fA3E4lgdBmy9Lc/edit?usp=sharing>

[Louay via remote participation, audio+slides]

Slide2. 360-video on HbbTV devices. One potential tech. No capability yet to render 360 video, so render in the cloud software and send to TVs, controlled by remote. Client needs only to be capable of playing livestreams. No API to control buffering.

Slide3. 3 different options. 1: all in the client, advantage=no dependency on network latency. Option2, our Hbb solution. Video buffering is a challenge, and each session needs server session. Scaling challenge. => Option 3a, pre-processing step happens once, then user input. Option 3b gets user-input later.

Use MSE to render 360 video in the browser.

Slide 5. advantages/disadvantages.

[video demo] requesting different segments for different fields of view.

We think native video players, getting URLs, can show 360 video.

Slide9,10. HTML VideoElement Extensions. Shows buffering switch between two views.

Laszlo Gombos, Samsung, Encode, tag, control 360° video

<https://pres.webvr.io/video.html>

VR browser, Samsung Internet for Gear VR.

You navigate to web page with video, play inline, and option to switch mode

That's not a great user experience, so we have an extension for the HTML5 video tag to describe how video is encoded. If browser detects that tag, it can go directly to the right immersive playback

[slide showing tags supported= what we found out there on the web]

This is how we've extended the web, let's discuss.

David Dorwin, Google, Path to native 360° video support

https://docs.google.com/presentation/d/1FYbOKq_CyUjoCztLPvzk49oRc9P6zsjGDgyo1aTrXHw/

Currently, non-rectangular media formats are not standardized, including projections.

Current solution: video->webGL->webVR. App determines and applies projection. Libraries in the near-term.

[slide3] Prereq: standardized media metadata.

Google proposal for ISO BMFF and WebM. includes custom projection

Recommended: Developing VR Media Technologies talk at Demuxed 2016

[slide5][slide6] Simplest approach, browser UX for spherical rendering, limited video-only experience

[slide7] More complete, make the DOM spherical.

[slide9] Encrypted media makes it more difficult.

[slide10] Media capabilities

Media Capabilities API on github, in WICG

Q&A

Kfarr: Another question, there have been a number of creative (yet competing) concepts for mixture of compression and projection optimization to reduce bandwidth. Facebook appears to be the most advanced in this realm so far, at least in terms of publishing their implementations and providing proofs of concept, however they have not open sourced any of this nor are they intended for use in WebVr / browser environments. While some mechanisms are too advanced at this point (such as foveated rendering based on gaze position), there are some that are clearly effective and relatively simple such as alternate projection styles and tile / block compression alternatives that offer significant bandwidth savings already. Instead of Brightcove

re-inventing the wheel to make yet another proprietary implementation thereof, can we work together to tackle the “known good” implementations as a standard?

David: we’re stuck with codecs today, experimenting with different projections.

Kfarr: Another question, If the “new” proposed “standard” were to include projection information in the mp4 container itself, there is significant existing content that would not have this tag. How would we deal with this transition period?

David: you can keep adding boxes to MP4

You can inject in JS. Ideally, we can get standardized

Container, you don’t need to re=encode, just repackage

natb@valvesoftware.com: in the interim, Samsung’s attribute proposal is good. Mesh extensibility is important.

Potch: FB’s pyramid approach and the approach we saw here use MSE and varying on field of view. Is that well specified in MSE today or do we need more?

David: for WebVR, you need sync between what the app thinks is the FOV and the stream. A bug link I skipped over, if you want to change the projection during playback.

Brightcove: How should we put the projection into the video tag? What are the literal strings that represent. Underscores versus dashes, how can we converge on one representation?

EricCarlson: Once we figure out what the strings are, they need to be added to RFC defining mime-type extensions. That’s not hard, we just need a proposal.

Klaus: precedence between media and tag?

@@: multiview support in VP9 similar to multiview HEVC?

Spherical DOM, what needs to be changed leaving cartesian space?

David: Important for W3C to figure out. New dimensions, CSS?

Klaus: Spherical DOM, per-element?

Potch: Circular DOM, CSS round

Casey: early in VR browser, tried to extend CSS Spec. Positioning and multiple orientations becomes confusing. Maybe there’s another place to do that.

BrianC: example video of Bill Clinton, camera in front of his desk, background was static. Rendering only from video file would make that more complicated. Also, matching of color between video and JPEG complicated

Rob: Camera capture, misleading to users to label 4k, effective viewport. Also are these sizes relevant if we have to scale to powers of 2?

cvan: eye tracking like with the FOVE, is progressive enhancement with http/2 or hinting something being addressed? also, is it a concern that the server could fingerprint with eye tracking?

natb@valvesoftware: prediction is complicated. It would be best to let people experiment. Decoding is complicated, mobile/laptop/desktop different capabilities. We're ready for high-quality 360 video.

Kieran: where do we go from here? What's the next step? Some complicated problems, some easy.

Dom: that's the landscape session at the end of the day. We'll have a workshop report shortly with standardization next steps.

Devin: propose a video representation breakout this afternoon.

Kieran: we want a standard plugin for videoJS. How do I get gaze position? Interactive video overlays+projection.

David: it's not so easy to solve short-term

Rob: Gaze position is really WebVR's pose - seems strange to replicate this at native level.

Wendy: W3C is actively looking for what's ready for standards, what needs more incubation. We'll have CG/mailling-list follow-up to keep you engaged .

David: apps can do today, jumping to a native solution may not be right before we understand the full range of the issues.

Justin: projection mesh means a very specific thing. Are people also interested in rendering state? E.g. blending, projections where pixels aren't directly rectangles we're transmitting.

David: metadata proposal not just for the web, cameras, etc.

Rob: If we're adding metadata related to cameras would be good to add camera intrinsics/extrinsics too (ala XDM)

David: custom controls vs app/default controls. Appeal of specifying in the dom is opportunity to overlay

Potch: custom controls different if I'm using video as a texture

A: we should be able to have controls on the second one

@@: WebVR with layer for video?

Justin: closed captioning and other accessibility features.

In Summary:

- Media type parameter for 360 videos (@IETF?)
- Container metadata for projection with mesh; include rendering state?
- Spherical DOM?
- MSE adapted to 360° video
- Expose Media Capabilities
- WebVR layer for video?
- Where to include close captioning & accessibility features?
- Ad XML (video ad placement)

Immersive Audio

Moderator, Olivier Théreaux, BBC; Panelists, Hongchan Choi, Google; Mike Assenti, Dolby; Raymond Toy, Google

Olivier: I work at BBC. I'm here today to make sure we're not missing half of the point of VR as Brandon put it yesterday about audio. We have a panel here to look at the status of audio & VR. We'll first get an update from Raymond, a lead developer on the Chrome Team and an editor of the Web Audio API at W3C, and will give us an update on the Web Audio Working Group, including the Web Audio spec which does synthesis and processing. The Working Group will soon recharter, so it is a good time to bring input on the next version of the specification. We'll have short presentations from each panelists, each followed by a short Q&A, but we'll also get more time after presentations.

Raymond: This is a brief overview of the Web Audio API. In the beginning, the Web was silent; you needed flash or a plugin to play audio. HTML5 added the <audio> tag, which allowed streaming and some limited functionality such as start, stop, seek. But you couldn't do much beyond playing and it had limited timing precision. 6 years ago, the Web Audio API was started to address these limitations and extend what you could do in the browser. It gives you an audio graph with sample-accurate timing with low latency. It allows you to create e.g synthesizer in the browser. Web Midi allows you to plug your keyboard into that procesing.

The Web Audio API gives you a way to define precisely when a source starts; in the audio graph, you get to use lots of different processing nodes (e.g. gain, audio effects, filters, spatial effects). Recently we have introduced the Audio Worklet that allows you to use JavaScript to do custom processing.

Where are we going? We have a draft spec that we are working very hard to bring it to Candidate Recommendation, hopefully by the end of the year. We left lots of stuff on the table to be able to move forward. We want to make the API extensible; for VR support, what do we need beyond what the API already provides with the panner? It seems to work pretty well already based on the demo I saw from Brandon, but we want to hear from you to make Web Audio useful for real immersive audio experiences.

Q&A:

Brandon: indeed, the room demo, it works really well. One issue we've found that if you turn your head fast, it creates discontinuity in the sound. What are your thoughts on how to improve this?

Raymond: that was an effect of the original panner node, but with the new API the problem should no longer exist.

Don Brutzman: in X3D, we have some experience in spatial audio from a content definition perspective. An [ellipsoidal front/back and linear attenuation dropoff model](#) was intentionally simplistic because most players were implementing this in software. Suggestion: consider setting the bar for an initial spatial-audio specification based on current hardware capabilities, and consider defining an even-high-fidelity spatialization for hardware folks to drive towards. Potential win-win.

Raymond: it turns out a lot of the nodes are based on OpenAL, but I'm not sure how to do more than that.

JamesB: question about codec support - Opus is supported in <audio> but not in Web Audio.

Raymond: that's something I'm working on fixing.

Olivier: next Hongchan Choi also part of the Chrome Team and member of the Web Audio Working Group, and describes himself as a Web Music evangelist. You wanted to talk about your spatial audio renderer, Omnitone

<https://docs.google.com/presentation/d/1Rrpv9kw18eIBUcllev84dKu94AQ9D4EeGusDEIEbg4Y/edit#slide=id.p>

Hongchan Choi: this talk tries to look at what you can today with the Web Audio API for spatial audio. When we looked at this in Chrome, there were 2 patterns: Object-based spatialization - we have a basic system for this in Web Audio with the Panner node

<https://toji.github.io/webvr-samples/06-vr-audio.html?polyfill=1>

The next proposal is to use ambisonics - a pretty portable audio format that can be played on any speaker setup. We project this into binaural rendering. But this was not part of the Web Audio feature set. We looked at how to make that possible.

One logical idea was to extend the <audio> & <video> element with a magical blackbox - but that seemed too risky.

Another approach was to use a MediaStreamSpatialAudioNode, but this had quirks as well. We'll probably go there at some point, e.g. in v2.

In the meantime, I started looking at doing this using the (now deprecated)

ScriptProcessorNode - it worked but with issues with latency and glitches.

But I took a step back, and realized I could do this using native audio nodes - and we got a great review from TechCrunch on the result which works in any Web Audio-enabled browser.

See googlechrome.github.io/omnitone

Some issues we discovered in the process:

- We're missing a compressed audio format for FOA/HOA multichannel audio stream; I think Opus solves this
- There are alternative spatialization techniques ground on native implementations

- Right now we have a separate rendering path for audio and video, which means there are synchronization issues that we need to address.

Olivier: we want to hear from the audience on what is needed for VR (whether the abstraction of the audio native nodes is right for the kind of needs of VR)

Q&A

Casey: are you looking at other spatialization techniques e.g. ray tracing?

Hongchan: not in this project; but it's a different matter for the spec

Chris: is this a worker friendly spec?

Hongchan: right now the Web Audio API is not accessible from a Worker, but we are developing this Web Audio Worklet

Chris: any high level library to wrap the API?

Hongchan: tone.js is pretty popular

Thomas: many 360 video comes with 2D sound - any way to make it sound better nevertheless?

Hongchan: I haven't thought about that

Philip: you mentioned orientation, position; I would have also expected velocity, also the room environment converted as a filter.

Hongchan: I was talking about the other approach - what you're talking is object-based parameterized spatialization. Ambisonics is not for that

Potch: any support for Doppler effect?

Hongchan: we had to remove it due to issues

Jason: I've done professional projects with Web Audio - great stuff. Being able to set the drop off rate would be useful.

Olivier: Mike Assenti from Dolby Labs will be looking at a different angle, on how to use object based audio for linear based VR experiences

Mike: I've joined the Web Audio WG recently, but been at Dolby for the past 8 years. I'll offer some perspective on audio content creation on audio for linear based VR experiences.

At Dolby, we distinguish interactive vs linear VR experiences. Linear is a storytelling or experiential (e.g. live sport event).

A soundfield is a description of sound at a particular point in space, whereas an object model-based representations describes semantically which sound (represented as a channel) comes from where.

There are dedicated hardware for sound field capture, but for object-based require a more onerous but more flexible artistic mix.

[illustrating with the diner scene of Pulp Fiction]

"Reality is not really what we want with virtual reality, but an artistic experience."

Live events - one of my favorite use case, you also need to mix the audio to make it pleasant.

You would capture this with different mic points. Doing this live is pretty complex (which companies including Dolby are trying to help address), but it gives you more artistic opportunities.

In a live music show, again, you want a curated mix, not a soundfield capture.

For post produced VR, you get all these sources that you turn into channels with metadata describing the objects (at least the position), which can then be rendered into the various speakers setups. This can be exported in audio formats that support spatialization - one possible export is to render this as a simple soundfield (although then again you lost flexibility). We talked about the difference between reality & VR. Rendering distance is not just a matter of attenuation - reverb also plays a role. You need metadata to determine e.g. if the object is supposed to be far or near.

Professional mixers don't want us to do the attenuation for them, but to get control of the primitives.

You can also do head-track vs non head-track audio (e.g. a commentator should be fixed to the user). You can also increase the gain based on the gaze direction.

In a live event, you might want to do automated panning based on the position of the artis on scene.

You could also mix VR in VR.

Speaker-rendering is also interesting to avoid the isolation aspect of VR.

What about the Web? We need 2 components to play this back: decode the content and their time-synchronized metadata, pass it to a renderer in which we pass the orientation & position of the user, with playback configuration (e.g. binaural).

The options to do this in Web Audio: we pass the bitstream to the native decoder tightly bound to the renderer, to which you pass orientation & position : but that latter part you can't do.

Another option is to the rendering in the Web Audio API (e.g. a series of panner nodes with reverb) - it gives more flexibility, but it's heavy and might get consistencies.

VR audio is more than head-track spatialization over headphones. Audio production is also art which need to be combined with ambisonics for great VR experiences.

Web Audio v1 brings us very close to linear based VR experiences, but not quite to the end - this is a feature request for v2!

Q&A

Philip: you talked about the linear stuff; what about the interactive stuff?

Mike: from a rendering standpoint, there is a lot of similarities. But at Dolby, we look more at linear content in general. But if we get it right for linear, it should apply similarly for interactive.

Don: a useful direction for future work would be for the 3D & audio to inform each other in VR experiences. For example, high-fidelity audio rendering can likely be accomplished for this room on the order of 100 polygons. Adding audio properties for reflection/absorption etc. is also analogous to visual materials/appearance. SIGGRAPH conference is a good resource for such work, e.g. [RESound: Interactive Sound Rendering for Dynamic Virtual Environments](#) by Manocha et al., UNC, 2009.

Shannon: My dream is to have a way to adding audio-reflective properties to the scene (e.g. to distinguish wood from carpet), and have the browser coalesce this.

Potch: from Slack: what does it mean to duck a sound field?

Hongchan: we have a compressornode that would enable ducking with side-chaining.

Potch: from Slack, have their been experiences in configuring the surrounding audio environment (e.g. getting more or less sound from your friends).

Mike: it's very much an artistic choice; it's still very early days in that regard.

Potch: is there an audio analog to photogrammetry?

Mike: there was a great paper on capturing multiple sound fields@@@

@@@ NFB: from an A11Y, how do we describe that audio?

Issues / summary:

- Compressed audio format for FOA/HOA multichannel audio stream
- Alternative spatialisation techniques and native implementation
- Tight synchronisation between audio and video frames (between all kinds of streams, audio/video, video/video)
- worker-friendliness and audioworklets
- Object-based audio pipeline - need to pass time-sync'd metadata ; sending orientation/position down to the decoder
- Capability reporting for rendering (important for graceful degradation)
- Speech synthesis
- Speech recognition
- Describing spatial audio for A11Y

Breakouts

 https://docs.google.com/document/d/1HxnxfzFvbigg_xH4180ROKH1eNNkgLWICpw_FhH1uQ/edit#

Standardization Landscape

Dom: We'll start our last session now. A couple announcements; a phone was found.

Another thing, we should do a group photo of the very first W3C Web VR Workshop.

We'll go outside and take the photo as a memory of the event.

Neil Trevitt: thank you W3C for organizing this. In July 2008 I met Bill @ at SIGGRAPH who said we should do a binding to Canvas and have eye glasses everywhere. We took it into Khronos and now we're here eight years later. Talking about standards landscape. Lots of interaction between what we are doing and what W3C is doing. Some of other standards orgs we come in contact with.

For those of you who don't know the background, we share the same ideals and processes as W3C; more common than different. We're committed to open standards and royalty free. We are focused on silicon software part of things. We have every GPU vendor. Our primary activity is to

figure out how to expose acceleration for graphics, computation and vision process with APIs and enable software to use them in real applications. We have about 120 companies.

Members keep joining each other [laughs]. How Khronos relates to AR and VR. We do file formats. Collada is an API. We talked about GLTF yesterday. Will save us all work and make applications more efficient and avoid siloed content.

WebGL is all on GitHub. GLTF Work is like a W3C Interest Group. We also have Working Group where people have signed up for RF. You can join and have a seat at the table. Core business is APIs for acceleration.

Timeline for where WebGL came from. Open GL in 2008 was ubiquitous. Open GL was available on all the desktops. OpenGL2 was becoming ubiquitous on mobile. So becoming available anywhere Web browser was running. Test message

2011 WebGL 1.0 launched. Four years not so bad. WebGL2 is based on ES3; four year pipeline or heartbeat.

New stack we have been talking about. Vulkan is the new.

Why is Khronos in WebGL? So closely linked to native API, silicon roadmaps. Portability; need GPU guys around the table.

What other issues do we have? Lots of the silicon guys - my day job is video - lots of VR capability, asynchronous context. Probably cannot expose all of them. Spoke to stereo first step. Do we want to expose more VR capability to expose more functionality at JS level.

Cameras, in 21st century we are still running processing on the CPU. If you can get power processing...ultimately orders of magnitude more. Need to get vision processing off the CPU.

Vision acceleration API. May not lift up to JS. Perhaps people implementing tracking might use native APIs like Open VX. PokemonGo is awesome but dispiriting to see that the power users must turn off the AR, because it runs better on the battery. Silicon has failed to enable applications for AR. Apache is available in the phones.

One potential effort is to use GPGPU in GL12. We're talking more general and higher level language and access to language not just for vision but also audio and neural nets. We'll want to accelerate that quickly.

We have WebCL; lift into JS; still kernels in C. Learn it's good when a standard takes off and

gets adopted, be thankful. Web CL was one thing we tried that did not work. Perhaps a JS extension is the way to go.

Good that the breakouts covered so much of this. The cameras are about to become really diverse in number; mobile phone has four. The wide angles, depth, stereo are coming. Just controlling the camera is going to be a point of fragmentation at JS and developer level. We have this nascent group called OpenKCam, intended to be an abstraction for how to control the sensor. This has not taken off yet; may not be needed. Maybe a cul de sac, or too early. Take a look at OpenKCam. Should we re-invigorate this group? Get lift into JS domain.

This is a cheeky slide: VulkanVR: is there a need? We're in a weird situation with VR SDKs. All similar but also different. Have apps running differently. Native rendering API things will consolidate quickly around Vulkan. @ has adopted Vulkan. UT, Epic, are porting onto Vulkan.

Seems the differences between these environments are not sustaining competitive advantage. Would it be good for that community building on all these SDKs not to have all that friction. Not do it all differently. If the native community gets its act together, that would make things easier to have more consistency at native level. Interested in feedback.

Slide with big chart of SDO landscape

MPEG guys, video, audio, images. They have done work on 3D compression. They do have a declarative AR framework called ARAF.

Then OGC deals with anything around geospatial. Tony, get Mark to look and they could help.

Here are my suggestions. We should meet like this much more often. Discover problems that we can solve with standards. Some don't need, but some do.

Figure out which SDO has closest domain expertise. Make sure they stay on track. Ensure community has a channel to feed requirements into the SFO. Continue regular meetings.
[Neil ends; applause]

Dom: I thought I would present what could be useful to standardize soon, longer term or never in W3C.

First is taxonomy. Most well known standards work is in Working Groups with formal IPR. We also have W3C Community Groups. WG only members; CG open to anyone. VR has been doing work in CG for now, has limited IPR commitment. We help to facilitate incubation. We also have Interest Group. Has more limited IPR.

Present what I heard, which is up for discussion. What should come up soon, later or not at all. And summarize what I heard over the past two days.

Existing relevant standards at W3C (slide 1)

Spatialized audio in Web audio WG

Gamepad API, Web Working in Web Platform WG

Media Streaming handling in HTML Media Extension WG

Low-latency data and AV transfer, identity hook in WebRTC

@@

Existing relevant standards (slide 2)

Color space management in CSS WG

Performance metrics in Web Perf WG

UI Security in Web App Security WG

Payments in Web Payments WG

New standardization efforts soon? [slide6]

WebVR is the elephant in the room. Gamepad API extensions. Some proposals

Notion of having a VR mode in CSS Media Query. Not sure if it's close or interesting. Broader conversation. Maybe something easy.

Notion of API that Samsung presented, putting context around content.

Has been a lot of work around Speech Recognition API in a Community Group

Interest in Web timing CG for Media synchronization; check out

Web Assembly is a CG at W3C. Still a lot of churn on this. VR might be another motivation to get done sooner v later.

Mention of proposal in a CG on Media Capabilities. May be too early and need more incubation.

[slide7]Longer term standardization targets?

Lot of discussion around declaratives around mark-up. An object model. Does that impact @ 360 media streaming we heard some approaches; open issues there worth looking into

Navigation Transitions; notion of metadata

Discussion around having a unified user input model for VR the way pointer events simplified touch. Maybe get something similar here

Gesture recognition framework may be further away; some cultural aspects; explore some primitives

Out of my league for Web fonts for 3D context

Several conversations touched on scheduling what happens very precisely. We have some mechanisms. Had some suggestions on how to inspire @

For accessibility can we imagine applying ARIA.

Annotating VR entities; maybe touch metadata

Identity and Avatar management is at the frontier between maybe and not at all

[Slide 8]

Help?

Things where I am unclear

DOM to WebGL?

What would need to be standardized for 2D-Web browsing in VR

We want innovation in this space but there is a cost; not have multiple viewports. Needs some coordination

UI patterns. We don't want to standardize, but will need to be some kind of agreement on how you interact with this space

Social discussion around binding of real and virtual worlds in terms of authenticity, geographic control. That is not necessarily ready for standardization, but worth watching

My proposal is that we extend by a little bit. You give us feedback on the priorities, what are the gaps. Idea is we use this white board to more easily brainstorm.

Rob: one thing missing is the permissions, especially in mobile. There needs to be a broader forum for discussion. Fusion brings APIs together. How to integrate as a whole.

Dom: Any other feedback on standardization landscape? Another interesting topic is there is somewhat clear separation between what SDOs do. Are there specific overlaps to pay attention to or carefully avoid. That would be useful.

Did we call out security explicitly?

Dom: Wendy, do you want to talk to that?

Wendy Seltzer: Thank you. Security is one of the areas W3C considers among our horizontal review areas along with accessibility, internationalization. We look at all the specs from this POV. Interesting to hear ways VR and Security intersect. About authenticity and integrity of environments. How do we give them indicators. How to deal with social interactions and recognitions. Great area to keep looking at.

Feature of device detection capability.

Dom: there was a media capabilities proposal. Something broader like head sets?

Olivier: Mention of media capability. There are broader questions, is this a 3D environment ready device I am using. Seems to be a need for that.

Dom: Capabilities. Any other input. What do you folks think we need to standardize tomorrow or should have standardized yesterday?

David: WebGL is blocking certain things you might prototype. Do it in WebVR

Dom: any ongoing work?

Tony: I saw a Chrome prototype from four years ago. That's been stuck in mud for years.

Brandon: this comes up a lot. We all need it but have higher priorities to work on like WebGL 2. Hope to give more attention to. Would be a nice to have. Was not so much of a pressing need.

Tony: That TripAdvisor demo I did. Do a rollover and you get a divet. I had to code every pop up twice. I just wanted an element ...

Brandon: I have made my WebGL voice known and said we need to do that now.

Dom: Any other input?

Anssi: Either people are tired or your proposals are near perfect

Tony: I'd like to talk about process

How to best participate; tooling, etc. We can use W3C infrastructure. What would you prefer? How many people use Slack? Pretty many. I guess Slack, combined with Neil's idea that we do this again sooner v later.

Dom: should we do another event like this a year from now? [hands go up]

Olivier: This felt like a weird workshop. Usually people come with 'I have a spec here' and kind of seeing where it sticks. This workshop a lot of topics but not a lot of new specs. So it sounds like six months time would be good where people come with these specs.

Wendy, W3C: I'm thinking about our strategy. We work well with proposals for concrete work. And we have places to help incubate things not yet at that phase. When we start a WG we draft a charter and get patent commitments. That is a great way to work on something that is ready to be specified and where we can describe to your lawyers what we are going to do. When we are not ready to do that, we have CG for incubation. We have discussions and repositories. And help discussions broader than W3C membership and look for when they are ready to become more solid standards. We're seeing there is lots of other activity and we aim to be good colleagues with other SDOs, where we can be helpful to others. I am excited by the energy and cooperation that we see here.

Shannon: talk about follow-up. February 27 or 28 there will be a WebGL meetup on VR. I have not yet posted the event. We'll have someone from W3C give us an update.

David: Seconding incubation and use in WIGC and GitHub. Start with an explainer and not go directly to IDL.

Phillipp: There is high performance graphics conference covering issues that we are doing here. Consider contributing to this. Will be be track on compiler technology. Call for papers will be in the next month or so, deadline in April. We invite not only academic but also industry. It's a great venue for a lot of the stuff we have talked about here.

Dom: Any last comments before we wrap up?

Anssi: Progressive enhancement or mainstream to look at? How do you apply this world? You have a large body of web content that is not VR.

Dom: Josh's demo yesterday was a good direction. It think it's time to stop. We have exhausted our brains and our energies. Thank everyone for coming. Thank the program committee, chairs, sponsors. Thank you Samsung for hosting. It would not happen without them.

The next concrete steps will be compiled in the report. Will be up on the website tonight or tomorrow. Let's keep in touch by email and Slack. My email is dom@w3.org. Reply to me to get back in touch. Opportunity for another workshop or event in some time.

[applause]

Time for photo!