Web AR in the wild: City Games and Communities

I thought that perhaps picking a project to conceptualize around rather than code would be easier for me as a way to get back into AR/VR and WebGL than (re)learning WebGL 2.0 and the WebXR Device API at the same time.

Something that has been in my mind a lot when thinking about AR is Daniel Suarez books: Daemon and Freedom™ and how can we leverage our modern technologies to build a next-generation networked community.

The idea of the "Darknet" as a communication hub and online community has always intrigued me. Think of it as an AR version of Lucasfilm's Habitat, a free-form unscripted version of our current MMORPG worlds.

The idea of making a Daemon-like networked environment using current AR technologies present several challenges, both social and technical that I hope to walk through in essay. We'll begin with the technical challenges, those are easier to articulate and present a lower barrier to understanding what they are:

1. How to get consistent outdoors AR?
    1. Does WebXR help with the placement of augmented elements in the real world?
    2. Are beacons (like Physical Web beacons, iBeacons or Nearby proximity API) a good way to introduce AR experiences
2. Can we get things other than markers to display out AR content from?
3. How do we translate 2D content to a 3D environment?
4. Can we move from 2D to 3D and back?
5. Can we generate multi user AR experiences?

For more details, watch Brandon Jones' presentation at Google I/O this year for more details on WebXR.

Before we jump to far let’s talk about WebXR.

What is WebXR

WebXR is an evolutionary development over the WebAR 1.0 and 1.1 APIs currently under Origin Trials, a way to run experiments for a subset of developers using a feature. This allows for rapid iteration and quick feedback but without running the risk of the feature becoming a defacto standard, particularly when the API or feature is not finished.

In this context, I’ll use WebXR to mean the WebXR Device API.

WebXR lets you create AR and VR experiences by providing access to input and output capabilities of AR/VR hardware.

For more information, check this Web Fundamentals Article that covers the API in more detail along with the rationale for the change.

How to get consistent outdoors AR?

If you’re using phones or other devices to work with AR experiences the issue becomes how to seed the environment. We’re not using devices like DayDream, Oculus or Vibe so we can’t have a full on VR experience, and that wouldn’t be the objective anyways.

Using the WebXR Device API we can now place virtual items in physical spaces

so users have access to them either through dedicated devices or through their WebXR enabled browsers using Magic Windows. The beacons could work presenting notifications to users that will then use browsers in ARCore/ARKit powered devices or with the right applications to experience the content as designed.

Can we get things other than markers to display out AR content from?

How do we translate 2D content to a 3D environment?

How do we move from 2D to 3D and back?

Can we generate multi player AR experiences?

Links and resources

- Inspiration
  - Daemon by Daniel Suarez
  - Freedom ™ by Daniel Suarez
  - Understanding the Daemon
- Groups
  - Google AR
  - Mozilla Mixed Reality Blog
  - Progressive WebXR
- Development Tools
  - three.ar.js
  - Playcanvas AR

        - three.xr.js
- Beacons and Placement Technologies
        - Physical Web
- Examples and Demos
        - webxr examples
- Hardware Devices:
        - Google’s Daydream
        - Oculus Rift
        - Samsung Gear VR
        - HTC Vive
        - Windows Mixed Reality headsets