Mixed Reality
Sense Smart Region Platform

Much like our own senses, sight, smell, taste, any additional senses are only useful if we can perceive and experience them.

Using Fiware platform, the Lora network and ingenuity of sensor enthusiasts, we can sense and gather more details that is not present to the human experience.

Augmented and Mixed Reality is gathering traction especially now Mobile Phones are powerful enough to use heavy 3D Image Based Rendering techniques to find contours and shadowing, and ascertain depth form parallax movement. (combined with gyro sensors on the phone itself.
However all Mixed reality sensing projects, have the same core issues.
- Where is the observer positioned.
- Where is the observer looking.
- Where are the subjects to observe
- How many subjects are there
- What information does the subject represent to the observer
- How to represent that.

Positioning issue

Currently the hololens has no GPS, it is a prototype designed for indoor depth and infrared scanning. LTU had plans to upgrade them.
Other mixed reality devices over time (like mobiles) will have this sort of information.

Positioning with GPS data is tricky due to line of site to the satellite grid gives better accuracy. So location with regards to being indoor, outdoors or surrounded by high buildings, can alter accuracy.

This means locating precise location of a real-world device that can be a 5 cm in size, would be extremely difficult especially indoors when GPS tracking is at it's worst.

Orientation issues.
The hololens does have gyros and can detection momentum and has a rudimentary compass. However the accuracy is no guarantee.

Rough solution,
So the assumption was we can still read the sensor network, and filter the devices at first by Distance to the observer and categorise them thusly.

We only need to concern ourselves with the accuracy of the sensors near-by, which could be solved using a tracking marker.

The project framework
- Connect to Fiware
- Collect the sensors. (JSON format)
- Organise and filter the sensor
- Allow for dynamic generation of expected tracking markers
- Render into the hololens

Unity, Vuforia, Hololens

Unity has a lot of integrations with both Vuforia and the hololens. Vuforia allows for tracking markers. And Unity allows us to deploy to the hololens.
We can read files from the internet and dynamically load in AssetsBundles from online storage (not included in this project). Meaning different "physical" regions could have specific assets to load (or unload).

Files And Projects.
You can find my work files over at my gitHub repository.
https://github.com/PaulDixon

I have tried to make the project in stages.

QR mockup

- https://github.com/PaulDixon/VuforiaQR
- demonstrate vuforia
- inside of the hololens
- with sensor data.

SSRP_classes

- SSRP_protocol.cs Classes based on sensor API protocols.
- SSRP_reader.cs : Connect to Fiware
    - checkInternetConnection
    - connectToSSRP
    - sendSSRPHandShake
    - CheckSSRPHandShakeResponse
    - sendAuthenticatedSSRPQuery
    - CheckSSRPQueryResponse

- - - SSRP_Connection_Error
    - readStoredResponseFromFile (I used File.IO initially, this was fixed in the SSRP_hololens version with the resource.load())
- Collect the sensors. (JSON format deserialize using Full Seriazer.
- SSRP_entity_manager: Organise and filter the sensor response spawn
- SSRP_context ellement controller (and it's associated view file) The class doubles as Sensor response and sensor context, since the response is the context element with some extra header information. So this class does the organising and filterering, and creates populates the pop up where we can

https://github.com/PaulDixon/ssrp_basic_classes

SSRP_vuforia

- Allow for dynamic generation of expected tracking markers
- Render the sensor data onto those markers.
- Nothing too exciting, click on edit player settings, XR settings, activate Vuforia. (assets will load). Drag and drop in AR camera, image from the same menus as create empy, cube, cylinder etc eetc

https://github.com/PaulDixon/ssrp_vuforia

SSRP_vuforiaDB

- Unity Package from Vuforia's Marker Database.
- API code and other Vuforia specific name/id details.
- Contains a branch of defunct Hololens code.

https://github.com/PaulDixon/ssrp_vuforiaDB

SSRP_hololens_src

- Clean Project with Hololens as it's core component
- Vuforia was added later
- Hololens specific export and configuration parameters
- Native hololens assets via holotoolkit, (it will reboot unity with extra menu's). Settup your camera, config and deployment preference, things like mixed reality camera, ray-tracing interaction points, gaze, stabilization, and gestures recognition.
- Unity 2017.4.x due to Far too many issues with legacy Unity5.x code
- Numerous bug fixes, restructuring of code due to timing or conflict issues between platforms and services, and usual frailty of the human mind (do as I want, not as I said).
- Used a singleton to link all the commands together.
- Used physical location of said singleton to gather all "Managers".
- Extremely messy work files as the Imported SSRP_vuforia prefabs carried additional transform meta information that made rendering their canvas's impossible for hololens (even if they rendered inside of Unity-Editor).
- A SPAM of canvases with incremental changes sizes, position, and cameras to render to, and distance plane inside that camera space place were rotated to find something that worked with the Holotoolkit mixed reality camera in Unity Editor AND

the hololens.

The rest is up to you ;)

Vuforia set up gotchas with the hololens
Vuforia configurations.

API key: get from account (see the readme in https://github.com/PaulDixon/ssrp_vuforiaDB)

		SSRP_vuforia	SSRP_hololens_src
Global	**API key**		
	Camera direction	Camera Default	Camera Default
	Mirror video	Background default	Background default
Digital Eyeware	**Device**	*handheld*	Digital Eyeware
	Device config	-	hololens
database	**(import but don't load or activate)**	-	-
Video Background	**Enable Video Background**	yes	NO!!*
webcam	**Disable vuforia play mode**	no	no**

creates a plane with Web camera texture that occludes the scene*
Vuforia will not initiate, and the projects will not function**

Player settings for building with

		SSRP_vuforia	SSRP_hololens_src
Other settings			

Configuration	Scripting Runtime version	Stable .net 3.5	Stable .net 3.5
Public settings	Capabilities	-	Internet client Blue tooth Spacial perception Microphone Web camera + Anything else
XR Settings	Virtual Reality supported	no	yes
	Enable Depht buffer Sharing	-	(NO causes matrix issues?)
	Vuforia Augmented Reality	yes	Yes *

This needs to be ticked even in the desktop for building ASSET BUNDLES*