



The Case for the Metaverse Browser

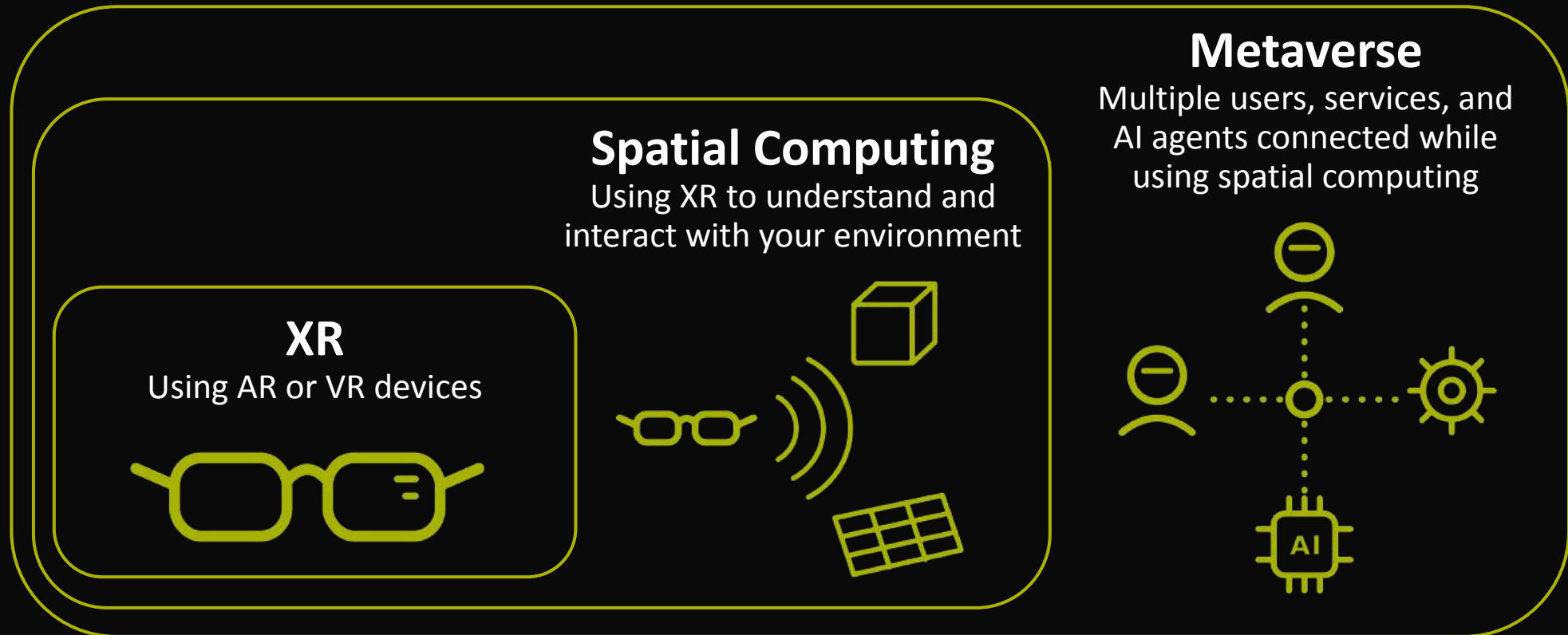
Metaverse Standards Forum | RP1

metaverse-standards.org/open-metaverse-browser-initiative

omb.wiki

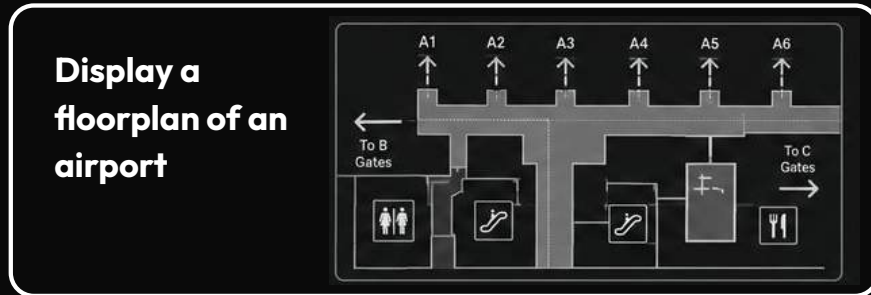
Metaverse Terminology

The metaverse combines the power of spatial computing with the connectivity of the web



The metaverse is the 3D evolution of the World Wide Web

The web was built for pages



Show a warehouse inventory report

SKU	Description	Location	Qty On Hand
TR-2301	Motor Assembly	Aisle 12B	58
DC-8873	Sensor Module	Aisle 28B	34
AA-1127	Control Unit	Aisle 14B	12
FS-3345	Power Supply	Aisle 24A	27
CM-7710	Connector Pack	Aisle 34C	73

The metaverse is built for **places**

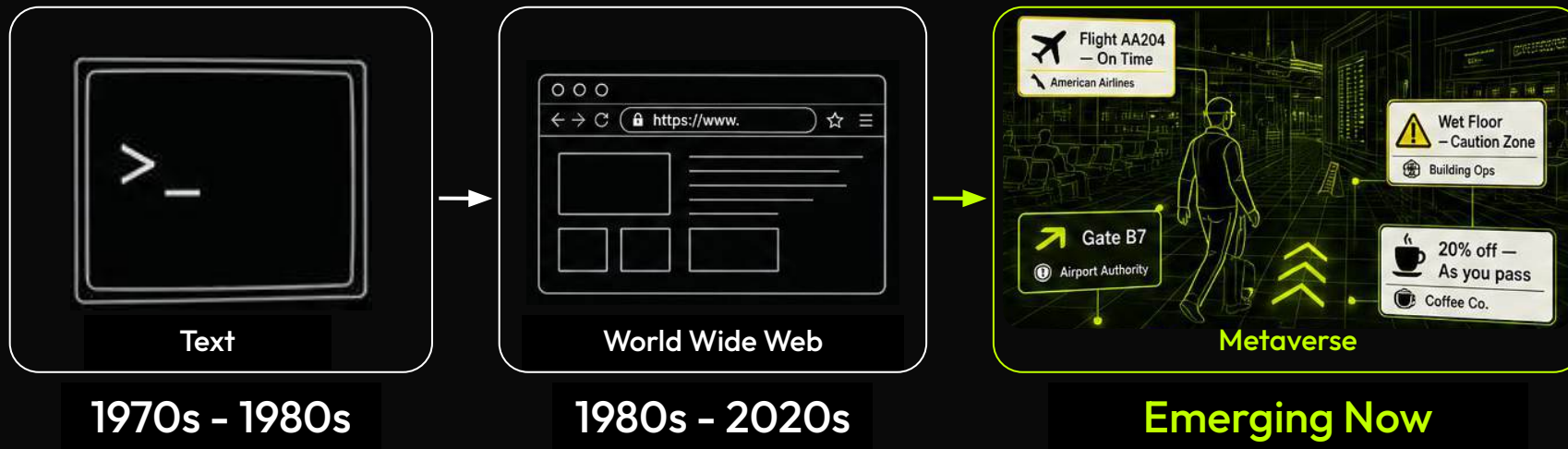


The world is going spatial

Every era of computing had its defining hardware



Each generation of hardware enabled a defining experience



AR glasses are racing to become the next mobile computing platform



Pioneers in AR Devices

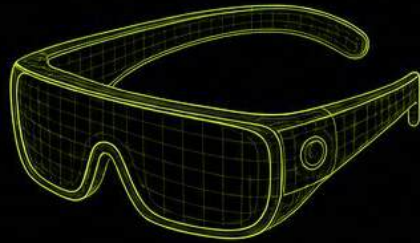
- Meta
- Snap
- Google
- Samsung
- Qualcomm
- XREAL
- RayNeo
- Rokid
- Even Realities
- & more

While mobile phones can handle spatial computing and physical AI today, AR glasses are designed to enable the optimal spatial experience

We need a global platform to deliver spatial services

Both AI and Humans are both going to need to connect to the real world

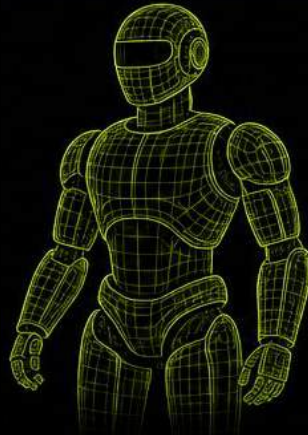
AR glasses
AI/Smart glasses
VR headsets
AI Wearables



Smart Homes
IoT Devices
AI Assistants
Digital Twins



Physical AI
Robotics
Autonomous Vehicles

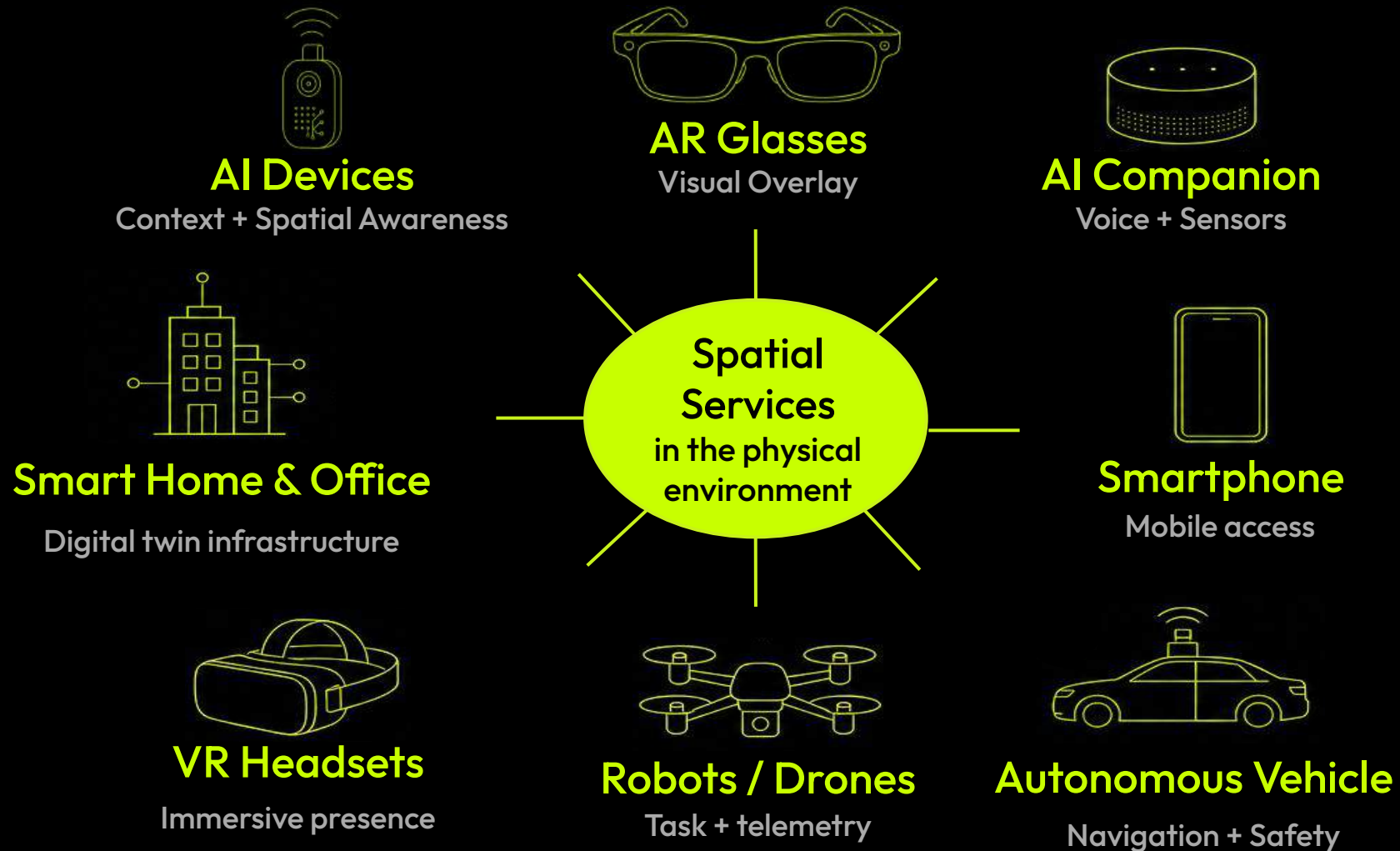


Citizen services
Smart Cities
Traffic Routing

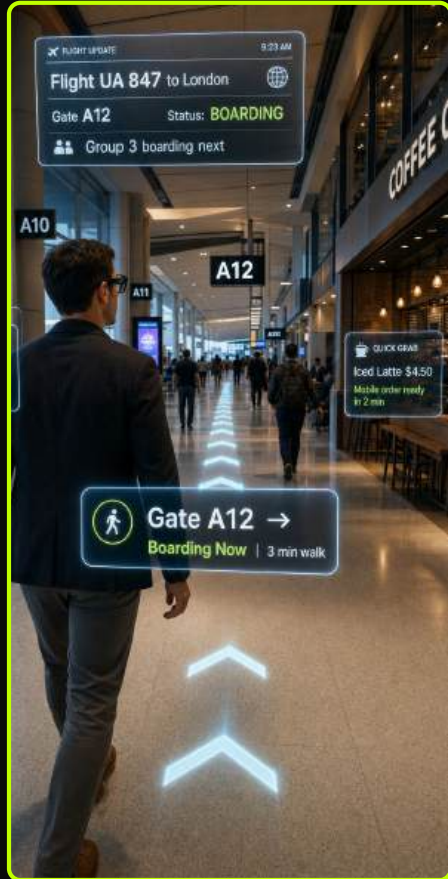


Spatial services will connect you to the world – not apps

Every device that operates in the physical world needs to discover and connect to services around it



Millions of **services** will run across the Metaverse



Airports

Navigation | Flights Status
Dining | Baggage Claim



Manufacturing

Machine Health | Quality
Control | Safety | Maintenance



Energy

Remote Expert | Pipeline Data
3D Schematics | Hazard Zones



Smart City

Transit | Parking
Events | Air Quality



Education

Lectures | Research
Campus Dining | Navigation



See our Open Metaverse Browser video at:
omb.wiki/introvideo



Joining from Seattle
Connected

The world is looking for how to
evolve the web to include
metaverse infrastructure

The web is the only platform with the needed
global reach and is based on open standards

Houston we have a problem!



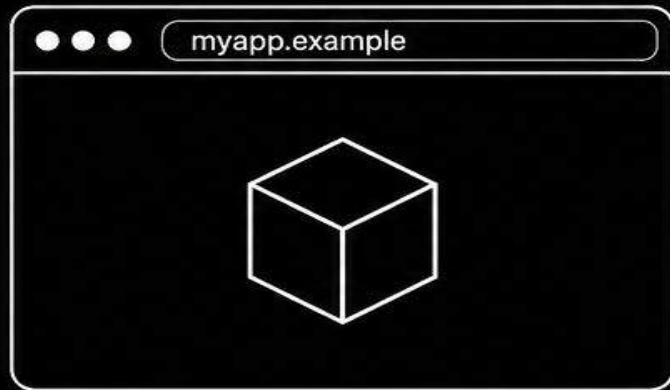
The web was not designed for spatial browsing



It's not a WebXR problem, it is a web problem

WebXR works for:

Single-origin apps



JavaScript → Canvas → one app, one origin.

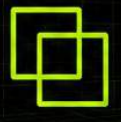
DOM and iFrames limit WebXR:



Can't easily dynamically connect to multiple discovered new services



Can't discover and load scenes and services based on proximity



Can't securely compose multi-origin scenes



Can't enable real-time spatial co-presence

Metaverse walled gardens exist largely because of the web's limitations



Hardware and software are vertically siloed

Each device ships it's own isolated app ecosystem



No network effect or interoperability

Content and users are trapped inside each platform



Everyone reinvents their own tooling, no shared ecosystem

Every platform, experience, and space is rebuilt from scratch



Unable to self-host spatial content

Can't host content the same way you host a website on your own server

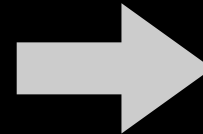
We're in the AOL moment for the metaverse and spatial computing

Moving from walled gardens to open standards

Walled Garden Platforms



1990s



Open Web Browsers

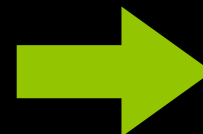


Walled Garden Platforms

Hardware & Software for Spatial Computing



Today



What is needed to unify the spatial computing industry?

The same way the browser unified the walled gardens before

Two worlds, neither complete

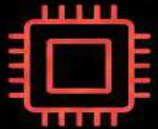
Each solves part of the problem. Neither delivers spatial browsing.

The Web (Browsing, not spatial)



Restricted performance

Javascript and V8 are not fast enough for high-performance real-time 3D



Restricted memory

Not enough for large or complex spatial scenes



Restricted device access

Camera, GPS, microphone beyond permission walls



Restricted storage

Cannot reliably store data locally through a web session

Native Engines (Unity/Unreal) (Spatial, but not browsing)



Requires compilation per device

Separate builds for each target platform



Updates are slow and manual

Must contact each device to update and reinstall



Not interoperable

Content built in one engine won't work in another



Not designed to stream

Large packages built to install and run locally

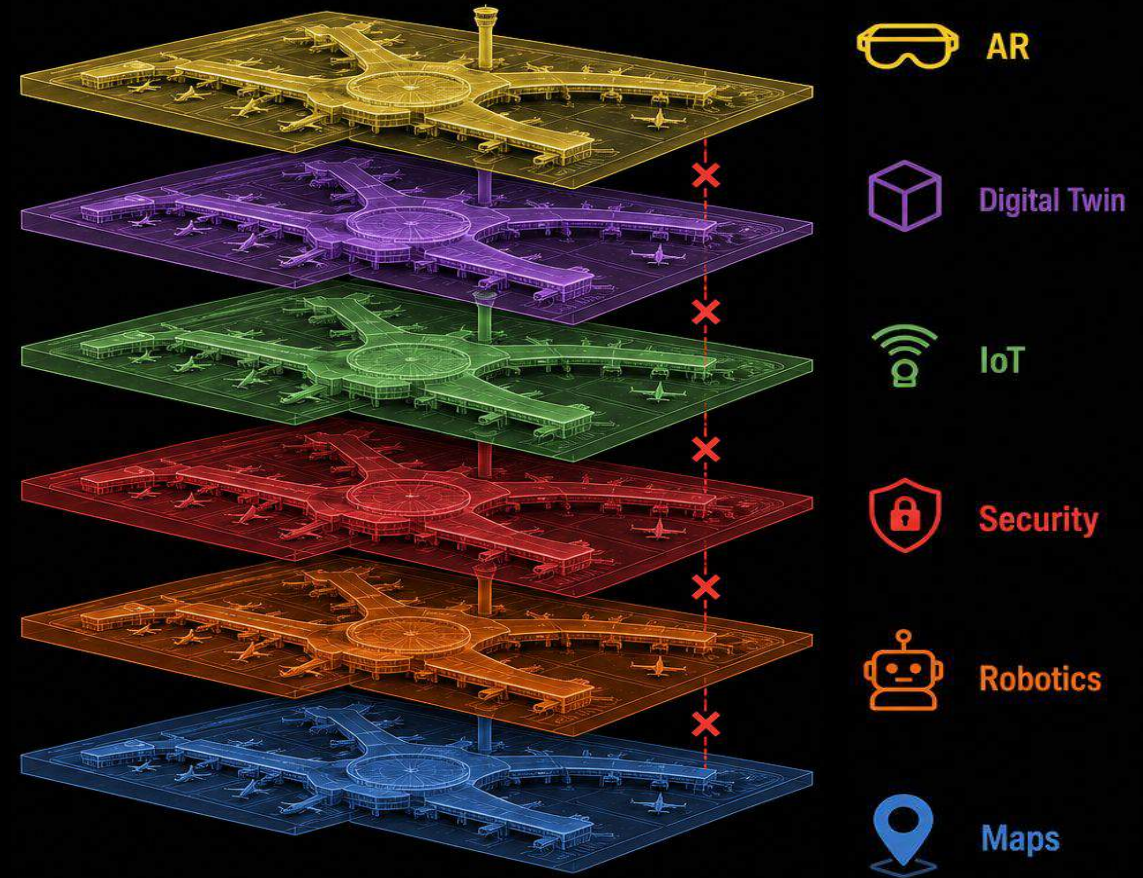
To get spatial browsing, you need both sets of capabilities in one place.

The industry is often fragmented

No interoperability. No unified access. Maximum friction.

- Every vendor runs its own maps, SDK, and data silo
- Robotics, security, IoT, and digital twins all operate independently of each other
- No System connects to any other
- AR experiences have no awareness of the environment around them
- No way to manage it all as one system

6 vendors. 6 separate systems. None connected.



Enterprises can't commit on infrastructure a single company can shut down

Single points of potential failure are not acceptable to many

~~AltspaceVR~~

Shut Down 2023

Community and content lost
overnight

~~HoloLens~~

Discontinued 2024

Enterprises stranded
mid-deployment

~~8th Wall~~

Acquired • Deprecated

Builders forced to start over

~~ReadyPlayerMe~~

Acquired by Netflix

Platform pivoted away from
creators

5 Enterprise Non-Negotiables to Deploy Spatial Services

The spatial computing ecosystem cannot reach enterprise scale without satisfying all five

1

Interoperability



Services built to open standards work in any compliant metaverse browser, regardless of device manufacturer

2

Device Agnostic



AR glasses, VR headsets, desktop, and mobile all access the same services from the same spatial fabrics

3

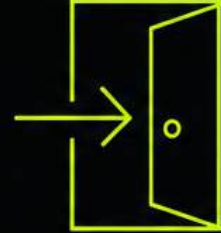
Data Ownership



Enterprises own their data, their customer relationships, and their spatial content. No platform takes a toll

4

Frictionless Access



AR glasses, VR headsets, desktop, and mobile all access the same services from the same spatial fabrics

5

Open Developer Ecosystem



Any developer can build and deploy spatial content – and build open tools for each other. No waiting on a platform to provide tooling. No permission required

These are enterprise procurement requirements

6 solutions needed that web browsers & closed platforms don't provide



Self-hosted spatial content

Closed platforms don't allow you to host your 3D content the same way you host a website on your own server. The metaverse browser connects to it directly



Publish 3D content like a website

Closed platforms don't allow you to drop content on a server and the metaverse browser engine retrieves, renders and displays it properly on any device (Mobile, PC, VR, AR)



Proximity-based scene loading

The engine loads content based on where you are. Proximity-based, built in. Relevant content appears automatically as you move



Trusted, unified multi-origin scenes

Independent operators compose one continuous 3D scene together. Each securely isolated from the others, all services in one view



Seamless service integration

Spatial services stream in and out automatically as you move. No stopping to install every service. No per-service setup every 10 feet



Real-time spatial co-presence

AR and VR participants share one space in real time. All devices (including mobile and desktop), one experience

**How can we meet these
requirements?**



Introducing Sneeze

The first open metaverse browser engine for spatial browsing



Embedded in current web browsers

- Powers standalone metaverse browsers
- Embed directly on XR devices

Open source (Apache 2.0). Built through OMBI under the Metaverse Standards Forum

Sneeze delivers spatial browsing

The reach of the web, the power of native apps, all in one engine.

The Web



Multi-device deployment



Instant access, no installs



URL-based navigation



Open standards ecosystem

Sneeze



Spatial browsing



Multi-origin composition



GPU-native rendering



Content-first development

Native Engines

(Unity, Unreal, etc.)



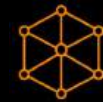
High-performance 3D graphics



Full device access

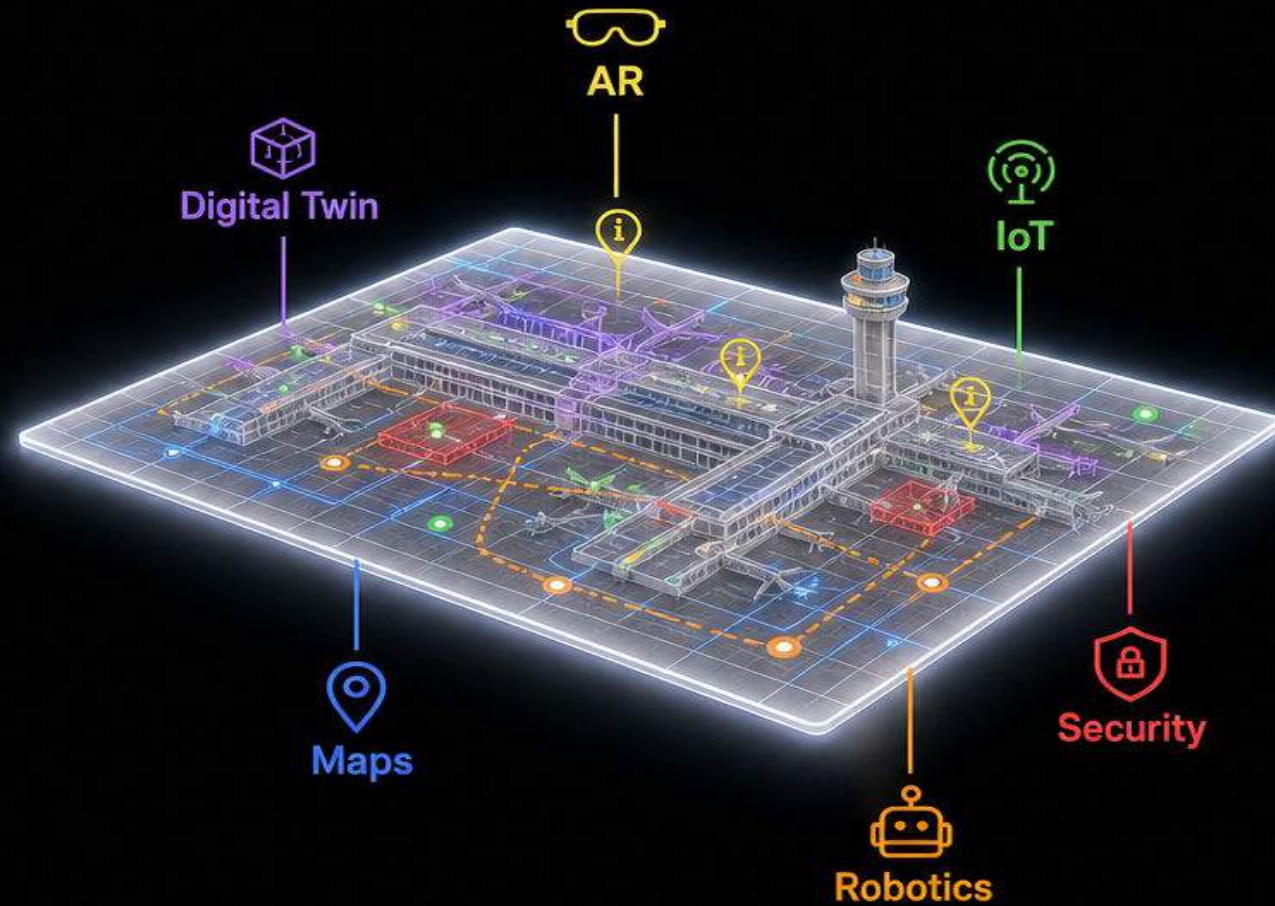
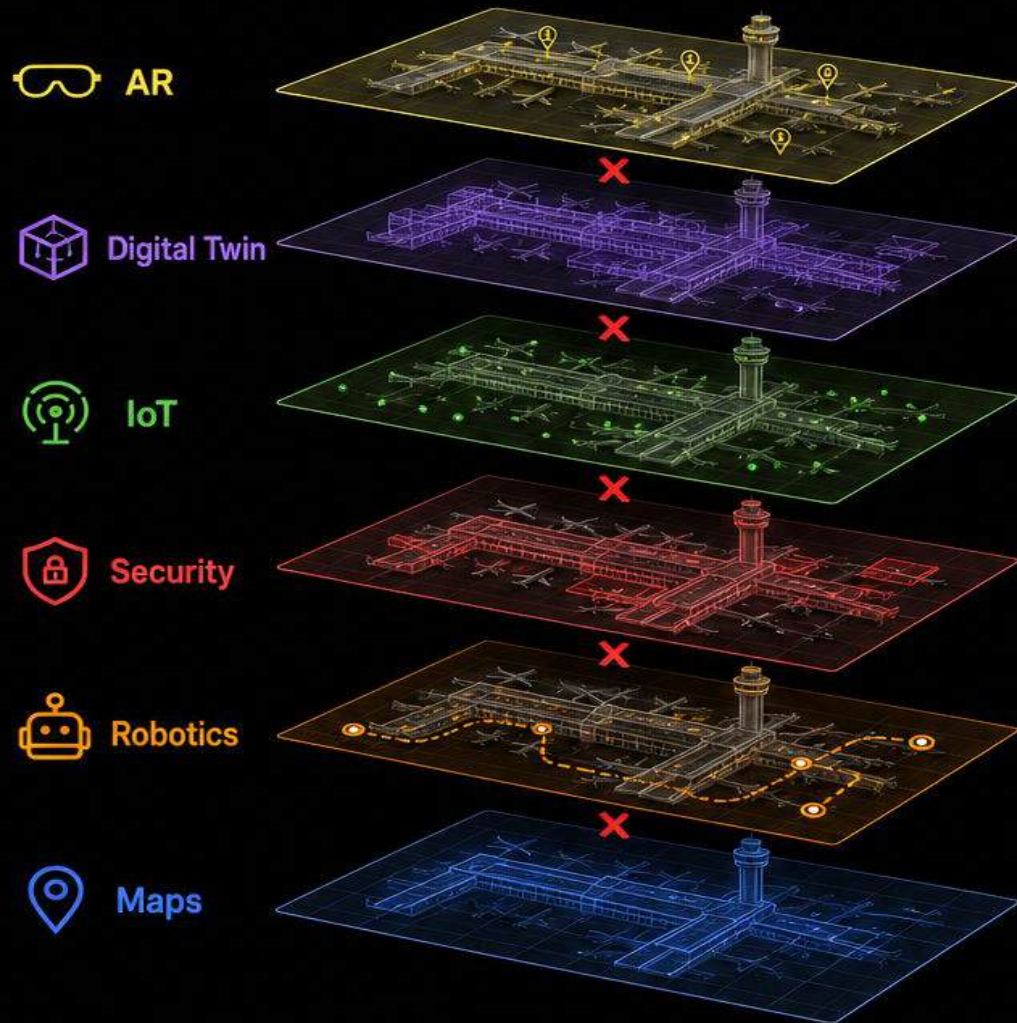


Large memory and storage



Rich spatial capabilities

Sneeze unifies industry fragmentation



Focus on your content. The browser handles the rest.

With Sneeze, you publish spatial content and the engine handles delivery, rendering, and display on any device.
Just like the web.

What Sneeze provides

Tangible capabilities the engine brings to the browser

01



Multi-Origin Scene Composition

Compose content from dozens of independent providers into one unified scene

02



Proximity-based Browsing

Connect to spatial services that stream in and out automatically – no installs

03



Spatial Anchoring

Physical and virtual world anchoring
Content loads based on proximity

04



Decentralized Identity

Cryptographic credentials
No per-service sign-on, portable across fabrics

05



Rendering Abstraction

Render to any device from one engine
(phone, AR glasses, VR headset, desktop)

06



Co-Presence

AR and VR participants share one space in real time – all devices, one experience

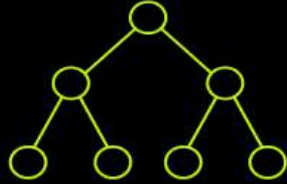
07



Certificate-Based Security

Signed certificates replace domain-based security as the basis for trusted multi-origin composition

Four key innovations to make an open metaverse possible



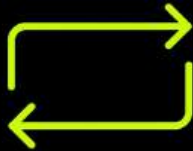
Scene Object Model (SOM)

Connected Services update a shared 3D scene graph in real time. The browser connects to and composes from multiple origins – no third-party platform or app needed.



Spatial Fabrics

A spatial fabric is the evolution of a website – a 3D map with anchored content and embedded services. It tracks user proximity relative to available services and offers connection.



RMAP

RMAP links the browser engine to spatial fabric services. Portable, performant, and robust service discovery and connection.



Certificate Based Security

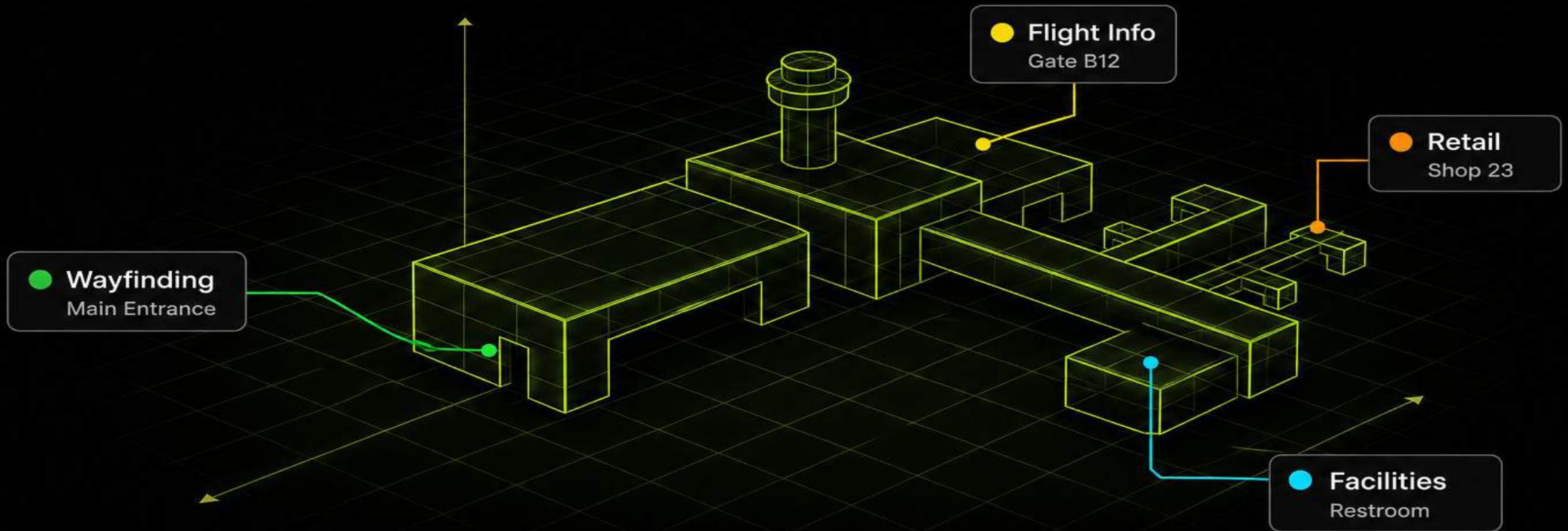
The web uses domain names for security, where Sneeze uses signed certificates for spatial fabrics which makes the SOM possible

Every concept that makes the web work has a direct metaverse equivalent

The Web		The Metaverse
 Web Server A server that hosts a website	→	 Metaverse Server A server that hosts a spatial fabric
 Website A group of web page served from one origin	→	 Spatial Fabric A 3D environment hosted by one operator
 Web Page A single page served from one origin	→	 Spatial Scene A 3D environment served from one or many spacial fabrics with embedded services at a location
 Web Browser Chrome, Firefox, Safari	→	 Metaverse Browser Any browser built with a Metaverse Browser Engine
 Browser Engine Blink, Gecko, WebKit	→	 Metaverse Browser Engine Sneeze
 HTTP / HTTPS Protocol connecting browser to web server	→	 RMAP (REST, WebSocket, Other) Protocol connecting browser to metaverse server
 DOM (Document Object Model) 2D document tree representing a page	→	 SOM (Scene Object Model) 3D graph representing a collection of spatial fabrics
 URL / Web Address Navigate explicitly by typing an address	→	 Proximity + URL Content loads as you approach

Spatial Fabric

A mapped coordinate space containing content and services



The spatial evolution of a website

Hosted on a metaverse server • Accessed by proximity

Spatial Fabrics: The metaverse equivalent of a website



Content and Services

gITF • USD • GeoPose



Composable

Many fabrics, content & services



Proximity Based

Enter a location to connect



Services + RMAP: Software embedded on spatial fabrics

Services are pieces of software embedded on spatial fabrics, from any provider.
RMAP is a protocol that connects applications in the browser to 3rd party services.



Embedded on Spatial Fabrics

Anchored to location in a scene



Self or 3rd-Party Hosted

Any provider can publish

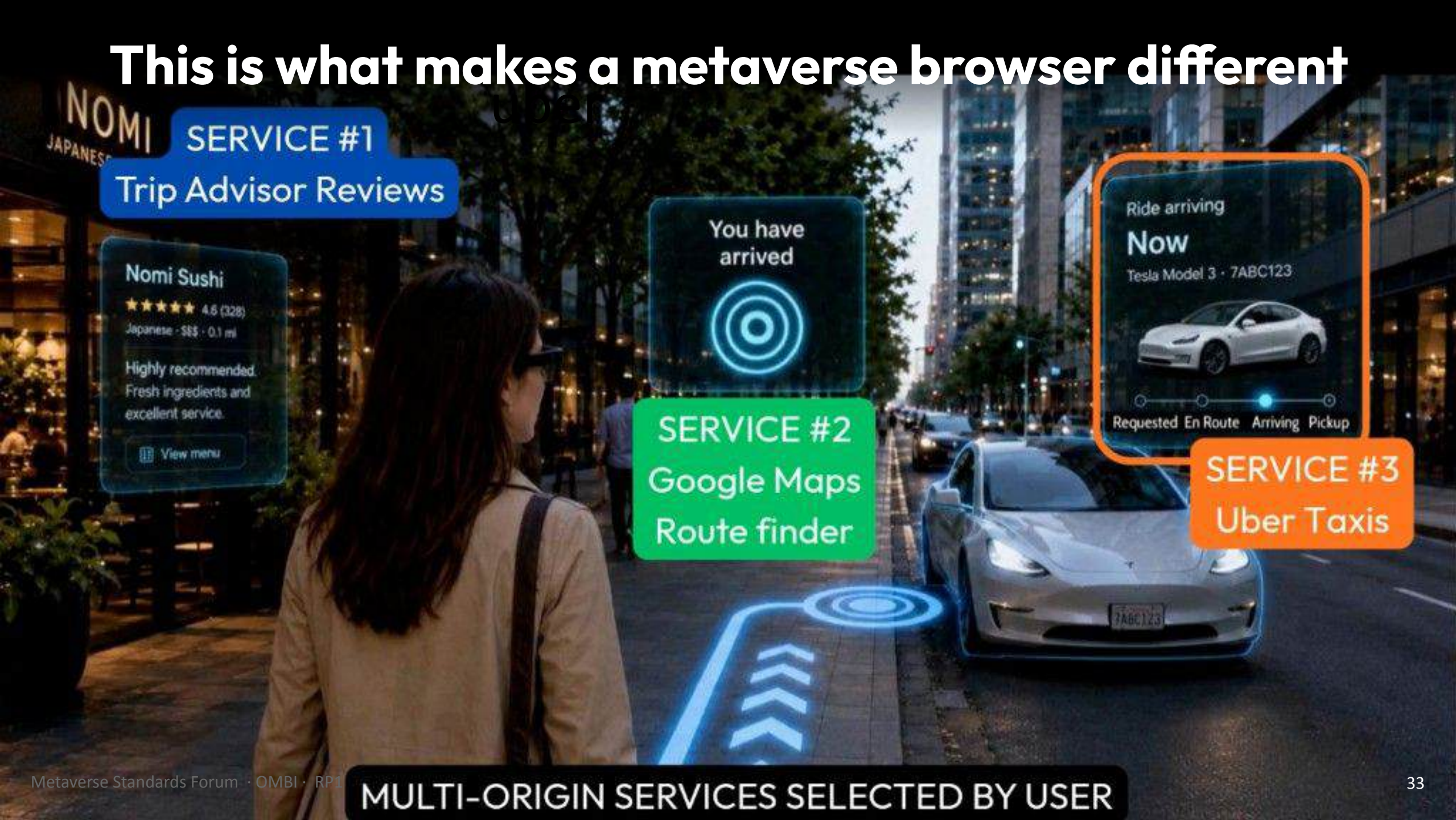


No Installs

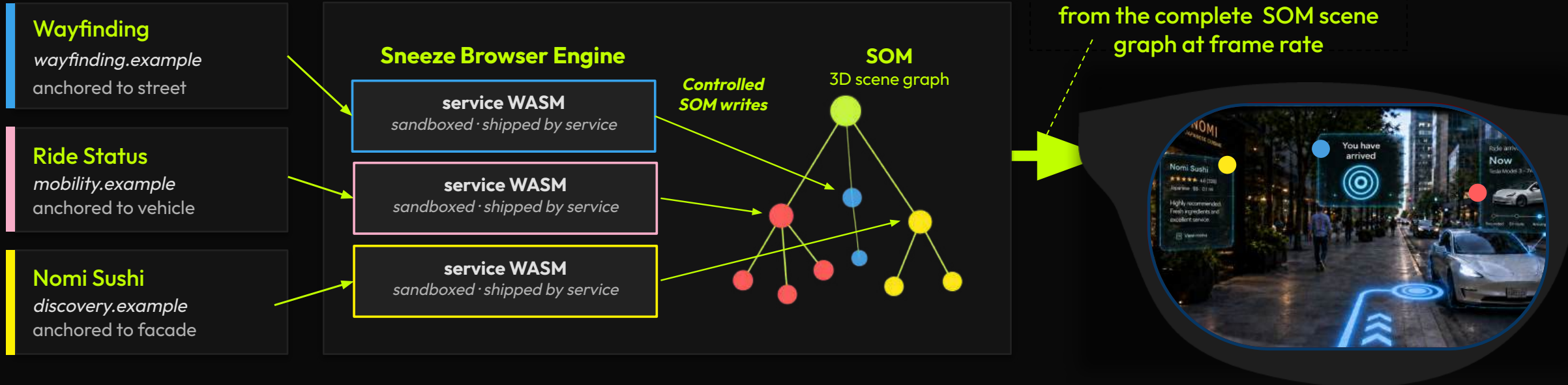
It works like visiting a website



This is what makes a metaverse browser different



How does the SOM work?



No app installs. 3 operators. 3 independent services. 1 continuous scene.

Each company hosts its own spatial fabric, controls its own data, and the metaverse browser composes them all — securely isolated

RP1 introduces the first *native* metaverse browser using Sneeze



Any device
Every OS

Powered by Sneeze
Built on open standards

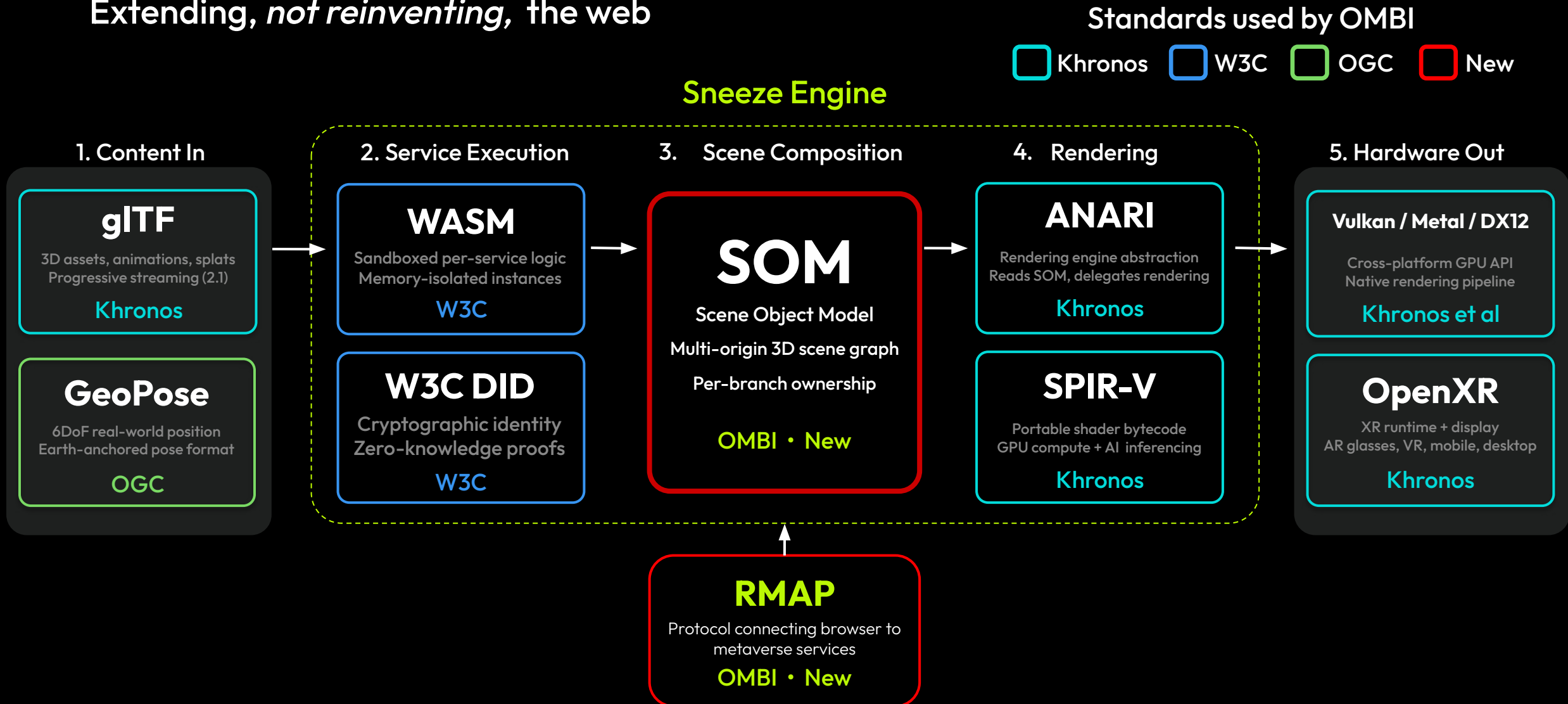
Many services
One scene

Next steps



Built on Open Standards

Extending, *not reinventing*, the web



OMBI Momentum is building

Open Metaverse Browser Initiative

Governed by MSF, enterprises are signing up and contributing

Launched • March 2026

Sneeze: Open Sourced

Open-Source Metaverse Browser Engine Launched

Launched • June 2026

3rd Party native metaverse browser

RPI's Artemis built using Sneeze

Launched • July 2026

Industry Voices on the Open Metaverse Browser Initiative

“

The most encouraging announcement of AWE



“

Big impact on the future of spatial computing

Forbes

“

Sneeze is a new engine purpose built for spatial computing

GamesBeat



University
of Rochester
Center for Extended Reality

ANNOUNCING

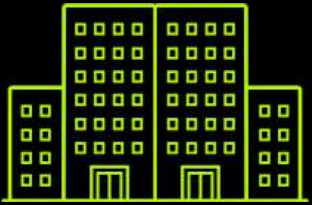
Open Metaverse Academic Alliance

Bringing together academic institutions worldwide
around OMBI interoperability for the open metaverse

With a University? Join Here

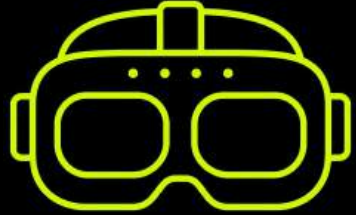


The world can start building now



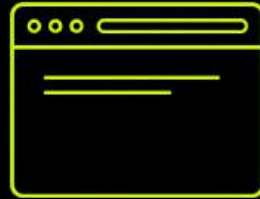
Enterprises

Build and deploy spatial fabrics on your own infrastructure



XR Device Manufacturer

Embed Sneeze and begin testing spatial browsing across devices



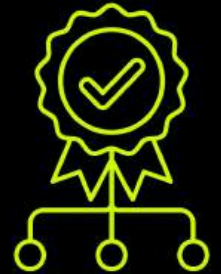
Web Browsers

Embed Sneeze and unlock spatial browsing to everyone



Developers

Build tools for the open ecosystem for themselves and each other



Standard Bodies

Create and extend needed open metaverse standards

The engine is open-source. The browser is live. Built on open standards.

Get involved

Join the open metaverse ecosystem



Join the MSF

Become a member of the
Metaverse Standards Forum.

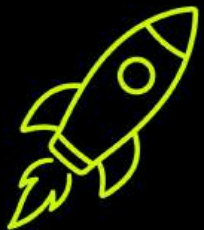
metaverse-standards.org



Explore OMBI

Architecture, engine docs,
and open standards.

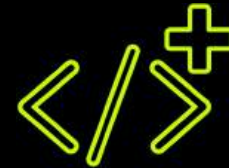
omb.wiki



Try Artemis

The world's first native
metaverse browser

rp1.com/artemis



Contribute to Sneeze

Architecture, engine docs,
and open standards.

github.com/MetaversalCorp

Let's connect

The metaverse is the next chapter of the World Wide Web.
Let's build it together.



Reach out:

ombi@metaverse-standards.org



Book a meeting:

rp1.com/meet



Metaverse
STANDARDS FORUM™



RP1®