

1. Why do we need standardized ORD protocols?

A critical obstacle hinders research into the metaverse - the absence of standardized Open Research Data (ORD) protocols. This limitation stifles the potential for exploring advancements in organizational practices within the metaverse and employees' working lives. We anticipate the Openverse initiative to accelerate VR research focused on organizational phenomena and employee behavior (e.g., virtual teams and leadership within the metaverse), which are currently underexplored (Aguinis & Edwards, 2014).

1.1. The Dearth of Standardized Protocols:

Organizations have contributed to the lack of interoperability in their pursuit of building virtual environments independently. Platforms like Horizon Worlds are inaccessible without a Meta account, and their software and hardware exemplify this issue, impeding data collection and restricting researchers' access (Ineqe, 2022). The absence of standardized protocols further obstructs replicability and reusability of virtual assets and data across diverse virtual environments. Addressing the need for standardized protocols becomes even more crucial when considering virtual reality studies within organizational settings. While these studies hold the potential to deepen our understanding of employee behavior in cyberspace, their impact may be stunted, and progress will be sluggish unless VR researchers universally adopt common standards and protocols. We, therefore, aspire to develop an open metaverse infrastructure bolstered by international protocol standards endorsed by researchers and industry leaders (Khorons Group, 2022)

In exploring diverse forums and websites of 24 different metaverse platforms, we casually observed the coding languages commonly used in these specific metaverses. The percentages corresponding to the usage of these coding languages are represented in the pie chart below, providing a visual summary (Figure 1).

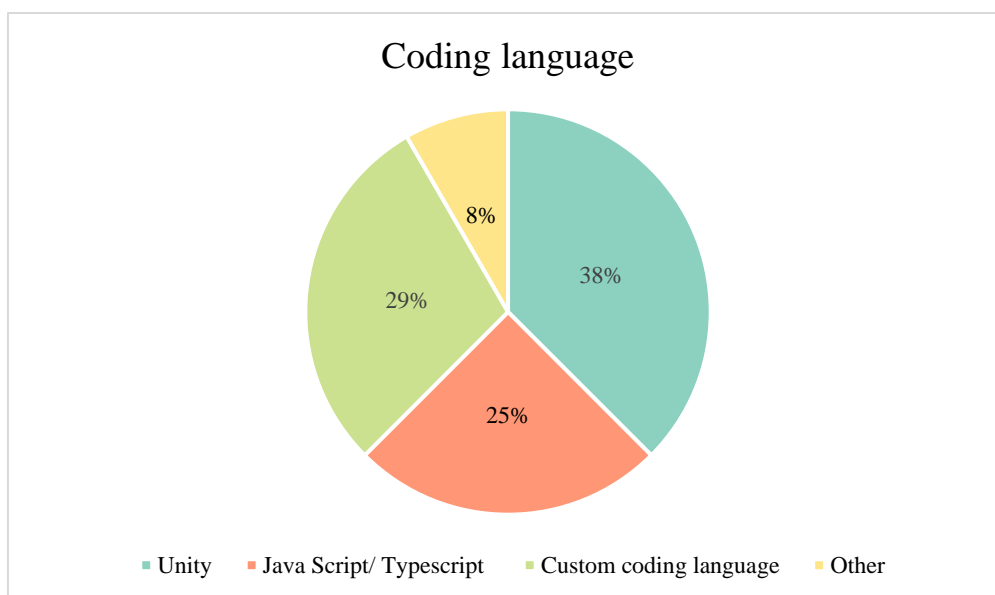


Figure 1.

Unity is the predominant coding framework among the surveyed platforms, constituting 37% of the total usage. Subsequently, custom coding languages such as Roblox Lua, employed in the Roblox metaverse, represent 29%. The remaining percentages in the pie chart are allocated to JavaScript and its extension, TypeScript, alongside other coding languages. A comprehensive table featuring all the surveyed metaverses is provided at the end of this report for reference (Table 1).

2. What is the aim of the Openverse to solve the problems that come with a lack of ORD protocols?

The Openverse initiative strives to fill the void by establishing an open metaverse infrastructure grounded in international protocol standards. Researchers opting into this infrastructure gain access to ORD-appropriate tools, enabling the collection, sharing, and utilization of virtual assets and metadata.

2.1. Benefits of ORD Infrastructure

The introduction of ORD protocols addresses the immediate challenges and empowers researchers. It allows them to conduct studies without the constraints imposed by corporations, ensuring ethical data collection and sharing practices. Virtual assets and (meta)data can be collected and made publicly available. ORD infrastructure becomes a cornerstone for fostering replicability, reusability, and the overall advancement of research within the metaverse. The Openverse spearheads this effort by initiating discussions among VR researchers to identify norms and coding languages. The goal is to ascertain the best practices for designing and operating virtual environments that adhere to ORD standards.

2.2. Ensuring Future Openness and Accessibility

The proactive establishment of ORD protocols during the metaverse's infancy is crucial. Openverse aims to steer the development trajectory towards the reusability and accessibility of virtual environments and data obtained. Encouraging interested parties to opt into the open metaverse and remain active fosters a collaborative approach, preventing the development of inoperable and incommensurable virtual environments. We want the open metaverse platform to become a self-reinforcing cycle requiring minimal upkeep as researchers add functionality and co-create virtual environments and employee experiences.

Table 1

Metaverse project	Company	Headset	Account	Coding language
Viverse	HTC	Vive (Focus), Oculus, Pico, and Steam VR	HTC Account	Unity
Microsoft Mesh	Microsoft Corporation	HoloLens and Windows Mixed Reality headsets (HP, Lenovo, Acer, Dell), Oculus	Microsoft Account	Unity, native C++ and C#

		Quest, HTC Vive Focus 3, Varjo XR-3 and VR-3		
Meta Horizon Worlds	Meta Platforms, Inc. (formerly Facebook)	Oculus Quest and Rift	Meta (Facebook) Account	Unity C#
Horizon Workrooms	Meta Platforms, Inc. (formerly Facebook)	Oculus Quest and Rift	Meta (Facebook) Account	Unity
Decentraland	Decentraland	Web-based VR, Oculus Quest	Metamask	Decentraland SDK, Javascript /Typescript
Sensorium Galaxy	Epic Games	HTC Vive Pro, Oculus Rift S, Oculus Quest and PC	Sensorium Account	Java Script, Solidity
Metaverser	Epic Games	PC	Metaverser Account	Verse
Roblox Metaverse	Roblox Corporation	PC and Mobile	Roblox Account	Roblox Lua (custom)
The Sandbox	Animoca Brands	Web-based and VR	The Sandbox Wallet/ Sandbox Account	VoxEdit (custom)
Somnium Space	Somnium Space	VR (Oculus, HTC Vive, etc.), PC, Web	Somnium Space Account	Unity
CryptoVoxels	Ben Nolan	Web-based VR	Ethereum Wallet	JavaScript
Altspace VR	Microsoft	VR(Oculus, Steam VR, Windows Mixed Reality)	Microsoft Account	JavaScript (Three.js)
High Fidelity	High Fidelity, Inc	VR (Oculus, HTC Vive, etc.)	High Fidelity Account	High Fidelity Scripting Language (custom)
Neos VR	Solirax	VR (Oculus, HTC Vive, Pimax, Windows Mixed Reality, Valve Index)	Neos Account	LogiX (custom)
VRChat	VRChat, Inc.	VR (Oculus, HTC Vive, Windows Mixed Reality)	VRChat Account	Unity C#
Sansar	Linden Lab	VR (Oculus Rift, HTC Vive) Windows PC	Sansar Account, Steam Account	Unity C#
Gather	Gather Presence Inc.	Web-based VR	Gather Account	JavaScript/ TypeScript
Rec Room	Rec Room Inc.	Oculus Quest, Steam VR, Pico 4, Playstation VR, PC, Mobile	Rec Room Account	Unity C#
IMVU	IMVU Inc.	Web-based, Mobile	Meta (Facebook) Account/ Apple Account/ IMVU Account	Core Website PHP, Client C++/Python, JavaScript
McKinsey Metaverse	McKinsey		Clients	Clients can use Hyperspace (no code platform) to create their virtual domains
Second Life	Linden Lab	PC	Second Life Account	Linden Scripting Language, Python, C++, XML, Mono
Axie Infinity	Sky Mavis	Mobile, Pc	Axie Infinity Account / Wallet	Unity
Upland	Uplandme Inc.	Web-based, Mobile	Upland Account	JavaScript
Everdome	Everdome	PC/Mac, Meta Quest 2	Everdome Account (Only for NFT holders)	JavaScript/ TypeScript, Solidity