

# Three Technical Challenges of Scaling from Social Virtual Reality to Metaverse(s): Interoperability, Awareness and Accessibility

Anthony Steed<sup>1</sup>

<sup>1</sup>University College London, Computer Science, London, UK

Correspondence\*:  
Corresponding Author  
a.steed@ucl.ac.uk

## ABSTRACT

The development of metaverse systems is marked by uncertainty regarding their technical architecture and the scope of their capabilities. Current social virtual reality (SVR) systems offer glimpses of future metaverse environments but remain fragmented, with limited interoperability, poor discoverability, and challenges in preserving assets and social capital across different platforms. Drawing parallels with the evolution of the Internet and World Wide Web, the essay argues that while certain metaverse standards may emerge naturally, fundamental challenges persist, particularly concerning client software. The essay highlights three overlooked yet critical issues for user experience: interoperability, scalable awareness, and accessibility. These challenges, often underemphasized in current standards discussions, are crucial for the practical development of user-friendly metaverse systems.

## 1 INTRODUCTION

There are many visions for what metaverse systems will comprise: from explorations of technical affordances of next-generation communication systems (21, 1, 32) through to rights and principles that should underpin such systems in use (23, 5, 3). If there is a consensus, it is that we don't quite know what the technical architecture or scope of capabilities of such a system will be: Will it consist of purely simulated virtual worlds or will it be grounded in mixed-reality spaces? What is the role of emergent network technologies such as peer to peer databases or blockchains? Who will control the systems and how will rights be enforced? How will the system be equitable to users with different backgrounds and access to interface devices? How will systems fulfil societal aspirations<sup>1</sup>?

We can see the early prototypes of future metaverse systems in today's social virtual reality (SVR) systems (e.g. see (22, 18)). These allow multiple users to congregate in diverse virtual places, share fantastic or realistic representations of themselves and undertake a wide variety of social activities. To some extent, a metaverse system should allow any existing SVR environment and activity to be experienced within a more unified framework where one didn't need multiple accounts, multiple devices, multiple pieces of client software and experience with multiple interfaces. In a metaverse we would expect to be able to reach all of the virtual places supported and we would expect fluid interactions with other users. Today, while

---

<sup>1</sup> E.G. See the European Commission's policy statement about the future of standards [https://ec.europa.eu/commission/presscorner/detail/en/ip\\_23\\_3718](https://ec.europa.eu/commission/presscorner/detail/en/ip_23_3718)

---

some users are already spending a lot of time in prototypical metaverse experiences, hardly any of the capital developed in one system (e.g. assets, but also social capital) are portable to other systems. Each is effectively a walled garden. Discoverability is low, with different experiences needing different software. Users are often not able to stay as a group within a single software client, never mind when switching to different software.

Today, designing a whole system that alleviates these frustrations, is an almost impossible task. Even the most evolved walled gardens of today lack significant features. Indeed some of the challenges about ownership, naming and scalability have already been identified in research going back over 30 years (e.g. see (24, 27)). There is a need for more systems to be prototyped that bring us closer to understanding the infrastructure components needed. One could point at other successful distributed engineering efforts such as the Internet and the World-Wide Web and claim that system will emerge by building on simple standards, and that relevant architectures will emerge from clear separation of responsibilities between clients and servers. But this belies that there are fundamentally difficult problems that client software that connects to metaverse(s) will need to solve, no matter their architecture. This essay highlights three such problems that affect the client software that a user engages with: what interoperability means; how to scale awareness levels; and how to ensure accessibility. These problems are described not because they are the most critical issues is designing future metaverses, but because they are some what overlooked by standards activities<sup>2</sup> and will have very significant impact on the experience of the user. They are derived from our own experience of teaching and developing for mixed-reality and also our experience in building platforms to support those activities.

## 2 INTEROPERABILITY

A metaverse system would necessarily be made up of client software that interact with server systems that provide critical features. While some of communication might be peer to peer between clients, as is done with some current game systems and research prototypes (e.g. see (27, 11)), within the services of metaverses we will find the distribution of description of virtual spaces, services for identifying and rendezvousing with other users, platforms for secure transactions, etc. One aspect of interoperability is that whatever configuration of systems we end up with, users will expect to be able to port of assets between systems(17). However, beyond that it is highly likely that dense social environments will require server infrastructure for simulation and message distribution. When we talk of interoperability, we thus identify that clear interfaces must exist for describing virtual spaces (i.e. are descriptive of scenes, containing geometry, materials, animations, etc.) but also for interaction between different processes across the Internet. Previous work on inter-operation has tended to focus on the latter, plotting architectures that build upon existing web services (e.g. (15)) or proposing new server simulation models<sup>3</sup>. There is an implicit assumption in a lot of this work: there will be a single integrated client that accesses these services. That is, if only these services could be described the client would take on the complete role of creating the user experience. Systems that have been built over the past couple of decades, have thus been largely based around this model of a large client that integrates all the interfaces necessary to connect to the back-end services. Second Life illustrates the advantages and disadvantages of this approach: while originally a closed system, the client was made open along with its protocols, so servers could be implemented to complement the official servers (for an overview see (2, 3)). Modern SVR systems are still walled gardens though their architectures will have some similarities.

---

<sup>2</sup> For example the Metaverse Standards Forum working groups don't obviously address such issues <https://metaverse-standards.org/>

<sup>3</sup> M<sup>2</sup> Morpheus Platform <https://codex.msquared.io/technology/m2-morpheus-platform>

While standardisation of services is certainly necessary, we highlight that almost all efforts are making this assumption that a single client is being supported. We note that with the rise of smartphones, consoles and tablets, the model of a single application taking control of the user experience has dominated. That is, while multi-tasking might be possible, certainly when it comes to immersive experiences a single client is responsible for creating the environments. Exceptions to this include the ‘holograms’ on Windows Mixed Reality/Hololens<sup>4</sup> or Volumes in VisionOS<sup>5</sup>. Both of these are examples of code from different sources that is integrated into a shared experience for the user. Notably both of these systems are primarily used for augmented reality situations, where the analogy of multiple applications matches a conceptual model of different objects in the world having different functions.

An interesting question is thus how far inter-operation between applications can be pushed. Rather than put all the behaviour code into a client that would then have to support code extensibility or interpreted code, why are we not treating the user experience more like an operating system or window manager, where different applications are responsible for different aspects of the experience? That is, how could we assemble an immersive experience from multiple applications?

Currently when we switch between different fully immersive systems each system takes full control of the user experience and is thus responsible for everything from input device interpretation, through simulation to rendering. Of course applications rely on run-time services for device interfacing, but there is little interaction between applications. This can be contrasted with mature desktop operating systems, where we not only have multiple applications and interfaces operating at once, but the screen is composited from multiple applications. Similarly, a modern web browser provides a variety of services over the operating system, with complex application being written within distinct pages in a browser that supports multiple windows.

There are precedents for this in research prototypes. The DIVE system was an early peer to peer multi-user system that supported a variety of immersive interfaces (11). The user interface was written in TCL/TK, so the client was dynamically extensible at run-time, but also applications could connect to a running client and insert new 3D objects as interfaces. This was an analogy of how X11 remote applications worked. A slightly different approach is the concept of scene-graph as bus, where multiple application interact using a common scene graph as a shared resource. The scene-graph acts as an interface between applications rather than using any other inter-process communication (31). We can imagine a variety of ways that different spaces could be constructed from volumetrically constrained, or through the window, metaphors for layering and composition. We can also see some hints of this in recent discussions about constructing mixed reality systems that compose real and virtual scenes (e.g. VRception (14)).

We therefore suggest that there are more ways that metaverse system could be constructed than simply constructing a large client browser. While efforts to extend web browsers through standards such as WebXR<sup>6</sup> to support larger collaborative immersive experience are very promising, e.g. see the Hubs project<sup>7</sup>, we believe that the key step is to re-imagine how an immersive interface is built. In Section 4 we will elaborate on one specific goal for client-side interoperability: supporting accessibility.

---

<sup>4</sup> <https://learn.microsoft.com/en-us/windows/mixed-reality/discover/hologram>

<sup>5</sup> <https://developer.apple.com/visionos/>

<sup>6</sup> <https://www.w3.org/TR/webxr/Overview.html>

<sup>7</sup> <https://hubsfoundation.org/>, formerly known as Mozilla Hubs

### 3 SCALE

For a metaverse system to scale to the audiences envisaged, there needs to be scalable systems that support persistent data to represent spaces, users, interactivity, but also scalable systems that support real-time synchronous interaction between subsets of known users. Fundamentally, real-time interaction between  $N$  users requires  $O(N^2)$  interactions between the clients supporting those users. Scalability to large numbers has long been an interest of developers from early distributed simulation days (e.g. see (24)). Early SVR systems used a variety of ways to determine whether or not to enable communication between pairs of users or entities based on cellular partitions of spaces (19), reasoning about spatial overlap of users (6), by partitioning a graph of nearest neighbours(4) or fixing group size of interested players (7). The general area is sometimes referred to as interest management (20, 9).

Many current systems use a partitioning of users based on a shard or instance models. That is, users are partitioned up into groups of users that can be supported on a simple server. Current SVR services supports small numbers of users ( $< 50$ ), as do many online games. Network measurement of existing systems (8) or open source examples (12) show that the server is a bottleneck: it has to ingest messages, simulate the world state and then distribute state changes.

There is thus a mismatch between the common vision of the metaverse as a massive online space, and the capabilities of the server and message passing infrastructure. But scalability techniques already note that user interest can be scoped down to the most proximate other users. Three levels of awareness have been identified (see also (27), chapter 12):

- Primary awareness: users are co-located and are proximate to each other. We would expect detailed animation, voice and joint interaction.
- Secondary awareness: awareness that a user is somehow in a reachable space, e.g. within the scope of a server or server network. This now means that rendezvous is possible and can be achieved easily, for example, by virtual travel.
- Tertiary awareness: awareness of the activity of activity anywhere in the population of users with which one is connected (e.g. friend networks), scoped by privacy policies. This to facilitate coordination and rendezvous of users.

Different systems might have additional levels, but it can be a useful exercise to map this to existing systems. Tertiary awareness is more like a social network or it might be labelled ambient awareness. The facilities provided are more about finding if a user is connected and available. The secondary awareness highlights that users are only reachable for primary awareness if they are in the same system. One can imagine tertiary awareness being an intra-system service of awareness, whereas to rendezvous one might need to install more software or download specific resources. Primary awareness is the full bandwidth experience. A distinction to secondary awareness is that in constrained situations, bandwidth and capacity is prioritised to support more relevant (i.e. proximate and important) users rather than users that are more in the background (e.g. far away or not engaged in the same activity).

These distinctions of awareness also highlight that in order to support travel in an apparently large-scale open world, multi-server support will be necessary. This then returns us to the question of partitioning load across servers, but now attempting to support seamless handover between servers. Again, research prototypes exist, (13, 16), but this is definitely an area that needs more research and development.

Finally, we will note that scalability will also have user interface aspects. Users want features to manage privacy and security, and these have to match with the practical routing of messages and distribution of

assets. Users also want to travel as groups and maintain cohesion. These present significant challenges to the network infrastructure as primary awareness should reflect user preferences, especially as collaboration is necessary to support certain accessibility features. Also, as soon as users make groups, some knowledge about that group needs to persist and potentially be migrated between different systems.

## 4 ACCESSIBILITY

Accessibility is an increasingly important topic in virtual and augmented reality<sup>8</sup> and thus it will be for metaverse systems (10). Previous efforts have covered aspects such as supporting users with low-vision (33) through to having one user act in support of a second user with restricted movement (28). These considerations pose enormous challenges to the future developers of metaverse spaces: how to support the broadest range of users without all developers having to support all features.

We again highlight that in other domains, such as windowing or smartphone interfaces, there is a variety of support at operating system and interface management layers that help make applications more accessible: from magnifiers to screen readers. Users will want tools that allow them to modify their experiences within the metaverse. Either as plugins to tools (as is done with web browsers) or through some other interoperability mechanism. While VRML97/X3D are no longer used to describe full systems, we would note that within the specification there is a clear responsibility for the browser to provide a set of interaction techniques for certain actions (walk, fly, etc.) but without specifying how these should be implemented<sup>9</sup>. Thus browsers have a lot of flexibility in implementation and can support customisation. While OpenXR<sup>10</sup> provides some functionality to abstract specific of mixed-reality devices, application developers still need to implement the full interpretation of device signals and generate a full-screen interface. Some decoupling of world description from basic interaction would allow users to use the devices that they need or prefer (25).

Finally, we note that some forms of accessibility can be facilitated by sharing of content. Indeed, in general with human-computer interfaces, different forms of collaboration around those interfaces is an important resource to support access (30). In recent years there has been a significant interest in tools that share immersive experiences, either through live streaming video, or more recently through re-sharing immersive experiences (e.g. see (29)). This touches on the other two themes of this essay: sharing content is an issue of supporting inter-operation between clients and sharing is one way of achieving scale by re-broadcasting. We should not think of metaverses as simple broadcasting systems, but as interfaces composed from multiple components. For example our recent prototype *AccompliceVR* (26) adds an overlay network for sharing single user VR experiences running on SteamVR<sup>11</sup> to remote users, and having those remote users appear as avatars superimposed in the original VR application. This was inspired by *Vermillion*<sup>12</sup>, which allows users to paint other SteamVR applications. Both applications exploit features of the SteamVR/OpenXR compositors and overlay systems which were originally designed to facilitate management of the device environment, application launching and immersive system controls. Such functionality hints at a potential interoperability route.

---

<sup>8</sup> <https://xraccess.org/>

<sup>9</sup> <https://doc.x3dom.org/author/Navigation/NavigationInfo.html>

<sup>10</sup> <https://www.khronos.org/openxr/>

<sup>11</sup> Valve Corporation, <https://store.steampowered.com/steamvr>

<sup>12</sup> Mountainborn Studios OÜ, <https://vermillion-vr.com/>

## 5 CONCLUSION

In this essay we have highlighted three challenges for future metaverse software. While current SVR software provides many delightful and useful collaborative experiences, there is no clear route to a system that could subsume the wide variety of features and capabilities that are expected. The challenges for any one developer of client software seem almost impossible to overcome with a single client: dealing with scale, interoperating with different services that will need to be implemented by different providers, while supporting accessibility to the broadest population. Our proposal is that we should rethink the role of client software, so that developers can focus on creating small components that are integrated by users or on behalf of users. By integrating lessons learned in other areas of software engineering for distributed applications, interoperability should be possible, which then enables scale and accessibility through allowing composition of systems and interfaces.

## ACKNOWLEDGEMENTS

This work was partly funded EU Horizon 2020 project Research Center on Interactive Media, Smart System and Emerging Technologies (RISE) (grant number 739578)

## REFERENCES

- 1 .Abilkaiyrkyzy, A., Elhagry, A., Laamarti, F., and Saddik, A. E. (2023). Metaverse Key Requirements and Platforms Survey. *IEEE Access* 11, 117765–117787. doi:10.1109/ACCESS.2023.3325844
- 2 .Au, W. J. (2008). *The Making of Second Life: Notes from the New World* (New York: Harper Business)
- 3 .Au, W. J. (2023). *Making a Metaverse That Matters* (Wiley)
- 4 .Backhaus, H. and Krause, S. (2007). Voronoi-based adaptive scalable transfer revisited. *Proceedings of the 6th ACM SIGCOMM workshop on Network and system support for games - NetGames '07* , 49–54doi:10.1145/1326257.1326266
- 5 .Ball, M. (2022). *The Metaverse: And How It Will Revolutionize Everything* (Liveright)
- 6 .Benford, S. and Fahlén, L. (1993). A Spatial Model of Interaction in Large Virtual Environments. In *Proceedings of the Third European Conference on Computer-Supported Cooperative Work 13–17 September 1993, Milan, Italy ECSCW '93*, eds. G. de Michelis, C. Simone, and K. Schmidt (Dordrecht: Springer Netherlands). 109–124. doi:10.1007/978-94-011-2094-4\_8
- 7 .Bharambe, A., Douceur, J. R., Lorch, J. R., Moscibroda, T., Pang, J., Seshan, S., et al. (2008). Donnybrook: enabling large-scale, high-speed, peer-to-peer games. *ACM SIGCOMM Computer Communication Review* 38, 389–400. doi:10.1145/1402946.1403002
- 8 .Cheng, R., Wu, N., Varvello, M., Chen, S., and Han, B. (2022). Are we ready for metaverse? a measurement study of social virtual reality platforms. In *Proceedings of the 22nd ACM Internet Measurement Conference* (New York, NY, USA: Association for Computing Machinery), IMC '22, 504–518. doi:10.1145/3517745.3561417
- 9 .Delaney, D., Ward, T., and McLoone, S. (2006). On Consistency and Network Latency in Distributed Interactive Applications: A Survey—Part I. *Presence: Teleoperators and Virtual Environments* 15, 218–234. doi:10.1162/pres.2006.15.2.218
- 10 .Dudley, J., Yin, L., Garaj, V., and Kristensson, P. O. (2023). Inclusive Immersion: a review of efforts to improve accessibility in virtual reality, augmented reality and the metaverse. *Virtual Reality* 27, 2989–3020. doi:10.1007/s10055-023-00850-8
- 11 .Frécon, E., Smith, G., Steed, A., Stenius, M., and Ståhl, O. (2001). An overview of the coven platform. *Presence* 10, 109–127
- 12 .Friston, S., Olkkonen, O., Congdon, B., and Steed, A. (2023). Exploring Server-Centric Scalability for Social VR. In *2023 IEEE/ACM 27th International Symposium on Distributed Simulation and Real Time Applications (DS-RT)*. 56–65. doi:10.1109/DS-RT58998.2023.00016
- 13 .Funkhouser, T. A. (1995). RING: a client-server system for multi-user virtual environments. In *Proceedings of the 1995 symposium on Interactive 3D graphics* (New York, NY, USA: Association for Computing Machinery), I3D '95, 85–ff. doi:10.1145/199404.199418

- 14 .Gruenefeld, U., Auda, J., Mathis, F., Schneegass, S., Khamis, M., Gugenheimer, J., et al. (2022). VRception: Rapid Prototyping of Cross-Reality Systems in Virtual Reality. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*. CHI '22, 1–15. doi:10.1145/3491102.3501821
- 15 .Havele, A., Polys, N., Benman, W., and Brutzman, D. (2022). The Keys to an Open, Interoperable Metaverse. In *Proceedings of the 27th International Conference on 3D Web Technology* (New York, NY, USA: Association for Computing Machinery), Web3D '22, 1–7. doi:10.1145/3564533.3564575
- 16 .Iimura, T., Hazeyama, H., and Kadobayashi, Y. (2004). Zoned federation of game servers: A peer-to-peer approach to scalable multi-player online games. In *Proceedings of the ACM SIGCOMM Workshop on Network and System Support for Games, NetGames'04*. 116–120. doi:10.1145/1016540.1016549
- 17 .Jerome, J. (2024). Faces and Places: Exploring Portability in Immersive Technologies. *SSRN Electronic Journal* doi:10.2139/ssrn.4739199
- 18 .Liu, Q. and Steed, A. (2021). Social Virtual Reality Platform Comparison and Evaluation Using a Guided Group Walkthrough Method. *Frontiers in Virtual Reality* 2
- 19 .Macedonia, M. R., Zyda, M. J., Pratt, D. R., Brutzman, D. P., and Barham, P. T. (1995). Exploiting reality with multicast groups. *IEEE Computer Graphics and Applications* 15, 38–45. doi:10.1109/38.403826
- 20 .Morse, K. L., Bic, L., and Dillencourt, M. (2000). Interest Management in Large-Scale Virtual Environments. *Presence: Teleoperators and Virtual Environments* 9, 52–68. doi:10.1162/105474600566619
- 21 .Oliveira, M., Jordan, J., Pereira, J., Jorge, J., and Steed, A. (2009). Analysis Domain Model for Shared Virtual Environments. *International Journal of Virtual Reality* 8, 1–30. doi:10.20870/IJVR.2009.8.4.2745
- 22 .Osborne, A., Fielder, S., Mcveigh-Schultz, J., Lang, T., Kreminski, M., Butler, G., et al. (2023). Being Social in VR Meetings: A Landscape Analysis of Current Tools. In *Proceedings of the 2023 ACM Designing Interactive Systems Conference* (New York, NY, USA: Association for Computing Machinery), DIS '23, 1789–1809. doi:10.1145/3563657.3595959
- 23 .[Dataset] Parisi, T. (2021). The Seven Rules of the Metaverse. <https://medium.com/meta-verses/the-seven-rules-of-the-metaverse-7d4e06fa864c>
- 24 .Singhal, S. and Zyda, M. (1999). *Networked Virtual Environments: Design and Implementation* (Addison Wesley)
- 25 .Steed, A. (2019). The Vehicle Pattern for Simplifying Cross-Platform Virtual Reality Development. In *VR Developer Gems* (A K Peters/CRC Press). Num Pages: 13
- 26 .Steed, A. (2024). Accomplicevr: Lending assistance to immersed users by adding a generic collaborative layer. In *2024 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)* (IEEE)
- 27 .Steed, A. and Oliveira, M. F. (2009). *Networked Graphics: Building Networked Games and Virtual Environments* (Elsevier)
- 28 .Thiel, F. J. and Steed, A. (2021). "lend me a hand"—extending the reach of seated vr players in unmodified games through remote co-piloting. In *2021 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)* (IEEE), 214–219
- 29 .Thoravi Kumaravel, B. and Wilson, A. D. (2022). Dreamstream: Immersive and interactive spectating in vr. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*. 1–17
- 30 .Xiao, L., Bandukda, M., Angerbauer, K., Lin, W., Bhatnagar, T., Sedlmair, M., et al. (2024). A Systematic Review of Ability-diverse Collaboration through Ability-based Lens in HCI. In *Proceedings of the CHI Conference on Human Factors in Computing Systems* (New York, NY, USA: Association for Computing Machinery), CHI '24, 1–21. doi:10.1145/3613904.3641930
- 31 .Zelevnik, B., Holden, L., Capps, M., Abrams, H., and Miller, T. (2000). Scene-Graph-As-Bus: Collaboration between Heterogeneous Stand-alone 3-D Graphical Applications. *Computer Graphics Forum* 19, 91–98. doi:10.1111/1467-8659.00401
- 32 .Zhang, Y., Kutscher, D., and Cui, Y. (2024). Networked Metaverse Systems: Foundations, Gaps, Research Directions. *IEEE Open Journal of the Communications Society* , 1–1doi:10.1109/OJCOMS.2024.3426098
- 33 .Zhao, Y., Cutrell, E., Holz, C., Morris, M. R., Ofek, E., and Wilson, A. D. (2019). SeeingVR: A Set of Tools to Make Virtual Reality More Accessible to People with Low Vision. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems - CHI '19* (Glasgow, Scotland Uk: ACM Press), 1–14. doi:10.1145/3290605.3300341