# L WHITEPAPER 1

METAVERSE–AS–A-SERVICE V 1.0

# 01
# VISION

## Decentralized System Services for the Open Metaverse

“[In the metaverse,] everyone sees the same stuff in the same place at the same time.”

– Neal Stephenson*

At LAMINA1, we believe that the metaverse is simply the future of all digital endeavors. We further believe that for the greater good, socially and economically, this network must be built in such a way that it is not controlled by any single entity or small group of entities — in other words, it must be an open metaverse built primarily on decentralized technologies.

*https://nextbillionseconds.com/2022/10/20/a-brief-history-of-the-metaverse-virtually-everyone/

The LAMINA1 blockchain is being purpose-built to power the open metaverse. But the metaverse needs more than a blockchain; it requires a system wherein all participants in virtual worlds can “see the same stuff in the same place at the same time,” with capabilities including real-time authoritative multiplayer synchronization, persistent state, avatars and user identity, spatial partitioning, object models, addressing and discovery, messaging, social networking, asset storage and delivery, and rendering.

To the extent that these services can be democratized and decentralized in a manner that is scalable, cost-effective, and equitably rewards all stakeholders, LAMINA1 aims to foster — with heavy participation from its community and partner ecosystem, and reliance on proven, best-in-class foundational technologies — the development and integration of turnkey, flexible, extensible and configurable systems to power the next generation of the Internet. As discussed in the LAMINA1 whitepaper, this forthcoming set of interoperating services to support virtual worlds will be known collectively as Metaverse-as-a-Service (MaaS).

In this whitepaper, we will cover various capabilities and designs to support this vision that are being developed internally at LAMINA1, including:

- The LAMINA1 blockchain
- General- and special-purpose LAMINA1 subnets (i.e. *metachains*) for flexible and scalable deployment of a variety of applications and experiences
- On-chain rights management integrated with high-performance multiplayer simulation and world state storage services
- High availability, reliable asset storage and delivery
- Early experiments in blockchain-based identity management

Additionally, LAMINA1 has identified several system services that we intend to work with partners to bring to market to support a thriving Metaverse-as-a-Service base layer and ecosystem, including:

- Real-time multiplayer simulation
- World state storage
- Secure decentralized user messaging

An ideal future state would be interoperable components for use by a range of metaverse creators and operators — from non-technical artists (writers, visual artists, musicians, filmmakers, etc.) to experienced programmers fluent in game development. Our mission in this space is to be a nexus for the creation of tools, open standards, and enabling services conceived and co-developed with a community of artists, builders, and service providers. As such, the Metaverse-as-a-Service model requires a diverse set of deep domain experts working together across many disciplines to solve some very hard challenges around how we create, work, and close critical feature gaps for those building the next era of online life.

This whitepaper explores the technical and business issues involved in such an undertaking and explores some potential solutions for system designs and projects to advance the endeavor*. The best solution we can put forth is a confluence of standards and verifiable working code. If we do this right, it will benefit dreamers, builders, and metaverse citizens alike.

**If you are developing protocols or infrastructure related to Metaverse-as-a-Service as described in this whitepaper, or want to provide feedback on this paper, please reach out to us at ecosystem@lamina1.com.**

* Note: This paper is focused on our Metaverse-as-a-Service offering and does not include additional information about tokenomics, carbon-negative or sustainability plans, provenance and large-scale payments, or various other LAMINA1 initiatives described in our original whitepaper. These topics will be covered in future L1 publications.

# 02
# THE LANDSCAPE

The market hype around the metaverse as a “next big thing” has largely ignored over three decades of technological innovation. In reality, the innovations that will power the open metaverse have been in development in earnest for a long time.

From networked, multiplayer real-time gaming to the rise of low-cost 3D game engines, to enterprise systems for creating and managing digital twins, many, if not most of the underlying pieces of the metaverse are here today — but nobody has put them together in the right way and at scale. The hype around the metaverse has galvanized the industry to take major steps toward interoperability, as evidenced by several new standardization efforts underway. But standards by themselves are no guarantee of success; we need to work together as an industry on products and infrastructure in tandem with standards, toward a common vision, with many principled points of view.

## 2.a Distributed Simulation, Real-time Rendering, and Multiplayer Gaming

Since the late 1980s, military “Vis-Sim” or “Distributed Interactive Simulation” pioneered decentralized simulation using state rollback, a technique considered state-of-the-art in modern Massive Multiplayer Online Games (MMOGs) today. Intelligence agencies seeded the world of “digital twins” by funding the Digital Earth, as seen by satellites and mobile sensors. And the video game industry has been driving the development of cheap and efficient supercomputer-grade GPUs since the mid-1990s.

A decade later, Second Life, Google Earth, and similar virtual worlds gained enough popularity that external development efforts began to standardize their underlying protocols. But these systems were limited in interoperability and, despite many peoples’ hopes and grand visions, never became the bedrock of a future open metaverse.

# 2.b Game Engines and 3D Runtimes

The previous two decades have seen the ascendance of game engine technology as a category. Before the 2000s, most game development projects required the creation of ad-hoc systems to load 2D and 3D assets, render images and scenes, handle user input, and perform other tasks to power game interactions and provide a satisfying experience — i.e. game engines.

As gaming grew into a massive global business, and as the technical problems became more well understood, developers increasingly turned to third-party game engine software. Two of the companies currently creating game engines, Unity and Epic Games, have become software super-providers, offering tools and services that power many of the biggest games in the industry. Their reach goes beyond games, as their products are now being used in many other industries, from film and television to military and medical simulations, to engineering and architectural design.

During that same time, specialized 3D hardware became commoditized and is now present in practically every consumer device, from high-powered PCs to game consoles, mobile phones, and virtual/augmented reality headsets. 3D rendering arrived in web browsers in the form of **WebGL***, the standard API for creating hardware-accelerated 3D connected web applications, and the open source libraries, such as **Three.js*** and **Babylon.js*** built atop it.

These engines and runtimes will power much of the metaverse — though they are not by themselves the metaverse; they are enabling technology. But they are ubiquitous and generally affordable. Combined with a new generation of tools, templates, and some of the novel network architectures described in this whitepaper, they represent foundational components of the open metaverse we, as a community, envision.

*https://developer.mozilla.org/en-US/docs/Web/API/WebGL_API
*https://threejs.org/
*https://www.babylonjs.com/

# 2.c Large-scale “Proto-Metaverse” Worlds

In the 2010s, Roblox, Fortnite, and Minecraft got closer to the conventional idea of “the metaverse,” and with a much larger user base. But all three continued the legacy of walled gardens, effectively holding users hostage in controlled environments. The benefit of Roblox and Minecraft, at least, was that anyone could create in it; for Fortnite, it was more about the fun of an engaging social experience integrated into the game world.

If you take into account the development cost of the underlying systems to support virtual universes like Roblox or Fortnite, creating one’s own is out of reach for nearly every developer. These systems were built with hundreds of person-years of effort. But with MaaS, anyone, in theory, will be able to make their own vast world like Roblox without having to make the astronomical investments made by those companies — because the entire industry is coming together to build the necessary infrastructure and share in the bounty.

# 2.d Early Decentralized Metaverse Services

Relatively new ventures such as The Sandbox*, Decentraland*, and Upland* are delivering innovative metaverse systems that offer transactions and virtual land backed by blockchain technology, under the idea that these could be independent of any commercial entity. These companies are now faced with hard (and to be expected) challenges around user growth — as we have learned time and again, it’s quality content and experiences that keep people coming back for more.

Even as these services find market fit via great content and use cases, they beg a few questions: What parts of these systems beyond land and item ownership are actually decentralized? And how will these systems scale? We can’t be sure yet, though it is very encouraging that companies innovating in these areas have made public commitments to standardization, and many are founding members of OMA3, a new standards group focused on solving metaverse interoperability problems.

*https://installers.sandbox.game/The_Sandbox_Whitepaper_2020.pdf
*https://decentraland.org/whitepaper.pdf
*https://www.upland.me/white-llama-paper

# 2.e Emerging Metaverse Standards

While the above developments represent antecedents and potential building blocks for the open metaverse, they have generally not been designed with interoperability and breadth of usage in mind — both of which are table stakes in the minds of many metaverse builders. To create future-proof infrastructure, there needs to be industry-wide cooperation and standardization.

There are several new standardization efforts underway, including: The Metaverse Standards Forum* incubated at the Khronos Group; The IEEE's* renewed efforts to classify and normalize metaverse concepts and creation of a Spatial Working Group*; The OMA* which is currently focusing on Web3-centric aspects such as ownership and world traversal; as well as the Open Metaverse Interoperability Group (OMI)*, Open Meta*, and other grassroots efforts. The World Wide Web Consortium (W3C)* has also quietly been pushing fundamental technical improvements for the web, especially in the areas of distributed identity, and is now creating draft specifications for viewing and interacting with 3D content directly in-browser without the need for additional JavaScript libraries or extensions.

For 3D assets like environments, buildings, vehicles, characters, and clothing, the industry already has two well-developed standards. **glTF*** has been optimized for fast delivery and lightweight mobile and web clients. **USD*** is designed for robustness and preservation of fidelity. But we need more. Looking at a single (and oft-cited) example, it is a very complex problem to make avatars that run everywhere; yet there is a pressing desire on the part of consumers to do exactly that. Formats like glTF and USD specify the 3D assets in a standardized way, but they will need to be extended to support robust behaviors such that avatars carry their own logic and interactions, independent of the worlds in which they're simulated, and potentially can be re-styled to accommodate use across worlds with varying art styles. **VRM*** is a popular extension to glTF specifically designed for handling 3D humanoid avatar data, with additions to the standard that specify body parts, body language, and facial expressions.

*https://metaverse-standards.org/
*https://www.ieee.org/
*https://sagroups.ieee.org/2874/
*https://www.oma3.org/
*https://omigroup.org/
*https://www.openmeta.xyz/
*https://www.w3.org/
*https://www.khronos.org/gltf/
*https://graphics.pixar.com/usd/release/index.html
*https://vrm.dev/en/

Though the interoperability of avatars is cloaked in all kinds of mystery, avatars are ultimately just specialized objects, and the open metaverse is going to need to get a lot of things right with respect to interoperable objects. Beyond content formats and behaviors, we need to think about ownership, provenance, and permissions. This is not just so that everyone gets paid; it is also to ensure privacy and security. This is why the marriage between traditional virtual world infrastructure and Web3 promises to be so powerful.

Interoperability is the key idea for both content and value creation. It's what adds the network effects that powered the web originally, back when interoperability simply meant text, URLs/hyperlinks, and slowly-loading JPEGs. But interoperability is not a specific end state; nor is it any one particular format or technology. Using the 2D Web as an analogy, there are many interconnected standards that power modern web experiences, and while some of them are fixed and fairly rigid today (e.g. JPEG), others are in a constant process of evolution.

There are multiple layers of sophistication to the metaverse requirements for interoperability. A little bit is a good start, but we need to get through all levels to really achieve the long-term vision. In the words of Neil Trevett, President of the Khronos Group* (the organization responsible for OpenGL, WebGL, glTF, and OpenXR), it is going to take a "Constellation of standards" to realize the open metaverse*. And it is going to take time.

LAMINA1 is a founding member of the Metaverse Standards Forum, as well as a member of working groups in several other standards initiatives. Our leadership has significant experience in developing open standards and open-source infrastructure. The systems described in this whitepaper are being designed in anticipation of future open metaverse standards. And as standardization efforts unfold, we hope to be engaged participants in helping to develop specifications based on our own learnings as we deploy Metaverse-as-a-Service with our partners and creators.

*https://www.khronos.org/
*https://www.forbes.com/sites/stephenibaraki/2022/07/22/2022-neil-trevetts-top-insights-chair-metaverse-standards-forum-president-the-khronos-group/?sh=242389c326b1

# 03
# DESIGN GOALS

Metaverse-as-a-Service is being built with the following goals in mind:

## Creator Centric

To bring the power of metaverse technology to any creator, regardless of budget and technical prowess, to foster creative and commercial success for the greatest number;

## Use-case Agnostic

To support a wide range of use cases, entry points, and devices* — because the metaverse is for everyone, not just gamers or people who own VR headsets;

## Diversified by Design

To foster a decentralized ecosystem of service providers without barriers to entry, where incentives are fully aligned among creators, builders, operators, and users, and control is in the hands of the people; neither LAMINA1 nor any other corporation or centralized entity should be in charge.

*https://medium.com/meta-verses/the-seven-rules-of-the-metaverse-7d4e06fa864c

# 3.a Target Use Cases and Creator Personas

The Metaverse-as-a-Service stack will target the following end users:

## Casual and Non-Technical Creators.

People who want an easy path to expressing their ideas, owning their digital works, and designing their own spaces. In the Web1 era, HTML and JavaScript democratized creation for the 2D web, and a groundswell of independents — some with no prior web experience — used “(right-click) view-source” as their way to understand how other sites were built. This openness enabled the explosion of Web content that made it the world’s dominant communication medium. But even with those relatively easy initial access points, web development was out of reach for many, giving rise to low- and no-code authoring and publishing solutions like WordPress, Wix, and Squarespace.

We should aim to parallel this success for the open metaverse, with an open stack that programmers and designers alike can use, and ready-to-go capability for those who don’t have the requisite technical skills. Moving beyond the 2D web, the metaverse also adds the need for a massive influx of 3D content, modeling and animation, environment art, and real-time, multi-user interactivity. There is plenty of software on the market that enables these capabilities: however,  it isn’t necessarily packaged today for metaverse creation.

Already, there are a variety of accessible tools to enable technical beginners to create or collaborate on things like setting up social venues, hosting group activities, displaying art in virtual galleries, and designing collaboratively, to name a few — and we expect this space to grow. As it matures and develops, we should foster sharing of code and knowledge in the style of the web in a way that allows anyone to “view source” for metaverse content, learning and expanding on what has come before.

# Independent Builders.

Metaverse builders who have the time and skill to invest into learning new tools. They may not be well-funded, but they're after big prizes — including free-to-play mobile games they hope to be played by millions, and branded virtual worlds that will impress top clients and keep them coming back for business. Indies need to save time and money, but they also want more control and customizability than no-code tools provide. On the other hand, they don't want to build a complex tech stack or instrument complex online operations;  instead, they rely on game engines and 3rd party libraries and services. Perhaps the majority of their development work consists of integrating existing elements.

Independent developers don't need a turnkey solution, but they do need batteries included. They also don't want to be locked into the biggest tool providers, because historically, they've found that those platforms constrain them. They want power, control, and freedom of choice.

To make developers' lives easier, LAMINA1 will provide and foster the development of feature-rich workflow tools and software development kits (SDKs) that support a variety of engines and services. This will help streamline the successful integration of LAMINA1 into future games, immersive worlds, digital twins, simulations, and other content.

# Established Developers.

Creators and operators of large-scale virtual worlds, massive multiplayer games, and large entertainment IPs. This user base may not need many of the services described in this whitepaper, as they have probably already rolled their own. However, they may be interested in the enhancements currently under development on the LAMINA1 blockchain that will help them tap into a vibrant community of creators and the rapidly growing user base of the open metaverse. By integrating with MaaS services, established developers would get easy access to the body of work that builders and developers across the industry are committed to making interoperable, and can actively contribute to it alongside millions of other open metaverse users and creators.

# 3.b Guiding Principles

The team at LAMINA1 is also focused on a set of core principles that will guide the development of the Metaverse-as-a-Service stack.

**1.Use the right tool for the job.**
Game engines have unlocked enormous power for millions of creators worldwide — but at the cost of usability and accessibility for non-technical and casual creators. No-code and low-code tools and scripting languages will provide a myriad of diverse creators access to the open metaverse.

**2.Manage complexity.**
Distributed data structures, compute and storage can be managed under the hood to create a uniform user experience at scale for many, if not most use cases; don't unduly burden the creator or developer within managing these unless they require explicit control.

**3.Separate concerns.**
The metaverse must be designed in a similar fashion to the web, where content, presentation, and interaction are decoupled. There will be many art styles, interaction modes, and delivery platforms — all of which can be powered by the same types of system services if properly designed.

**4.Decentralize pragmatically.**
While decentralization is a primary goal, it cannot be dogma. At this point in time, many metaverse services are far more efficient and/or economical when they are centralized. At the very least, creators should be offered choices of services and make their own decisions. The perfect should not be the enemy of the good.

**5.Offer the appropriate model(s)
for the experience.**
Let creators create, players play and businesses work as they see fit, without forcing business models like artificial scarcity, play-to-earn, or walled-garden access when they are not warranted. Metaverse-as-a-Service needs to be just that: a service, not another yoke around creators' necks.

**6.Let services be 'of service.**
We intend to provide a framework for all kinds of services — even ones that compete with us or replace us — to find a place in the market. Customers can choose which and when to use each of these services.

**7.Keep the economics simple.**
If your users don't easily understand how something will make them or spend their money on a daily basis, they shouldn't use it — and you should find a simpler model.

# 3.c An Open Process

Fundamental to the development of the open metaverse is an open process. This whitepaper is part of that: LAMINA1 technical leadership intends to conduct its thinking out in the open at every stage of development, and we encourage our readers and community members to get involved in the Metaverse-as-a-Service effort as reviewers, contributors, technology providers, and builders.

We fundamentally believe that the open metaverse won't happen unless it is through collaboration across the industry on technical standards, thought leadership, customer conversations, and full-throated support of creator rights backed by action. Unless these activities take place in the light of day, the danger of walled gardens emerging once again is simply too great.

**glTF** is an illustrative recent example of a very successful, fully open development project. Though only Khronos voting members can ratify specification changes, anyone is allowed to contribute to the specification regardless of membership status, and the entire specification effort takes place publicly on GitHub via issues. In the course of a decade, glTF went from a small group of energetic contributors to a global effort and now, an ISO standard used by millions.

LAMINA1's principals have extensive experience developing standards and leading open-source projects — including glTF. Our view is that the path to success in the Open Metaverse is going to be a winding one, involving an interplay between design, experimentation and prototype implementation, commercial deployment, and eventual standardization. Our designs in this whitepaper reflect that point of view.

# 3.d Powering the Open Metaverse Economy

Imagine a future where the metaverse is open to not only every creator but also to any size and type of technology or service provider. Envision, too, that every creator has freedom of choice as to who hosts their content and operates their live spaces — be it a central service or a set of federated operators. And further, that they can have their pick of additional services, tools, and content templates to make their places in the metaverse unique and special.

Much like the 2D web, this is where the metaverse needs to go. Think about an open marketplace of services to power the networks, provide the asset delivery services and running simulations, and offer a range of tools. It would look a lot like a web ecosystem: ISPs provide connectivity; WordPress and other CMS systems power the content; and plugins add capability e.g. e-commerce storefronts, and music players. The open metaverse will have analogs here, and there is going to be massive opportunities for a breadth of Metaverse Service Providers to participate in this economy of the future.

Contrast this with the current state of affairs, where massive multiplayer games and proto-metaverse worlds control the entire tech stack and run the economy. There is no point of entry for small technology and tool providers. It's also not robust, as it has a single point of failure.

In the future metaverse, analogs to today's WordPress and ISPs could come in the form of simple template-based 3D world creation tools — from out-of-the-box metaverse servers that are able to support a small number of users (e.g. up to 100) via P2P services that spin up at the touch of a button, to turnkey large game worlds that can support millions of users. In fact, proto-metaverse services like these already exist in one form or another today, but not in a modular and flexible way.

For example, OnCyber provides an easy way to display NFT art in an art gallery walkthrough. However, the tools and hosting service are bundled together and controlled by a single entity. In the open metaverse we are envisioning, the gallery creation tool would be separate from hosting.

The key to unlocking this future is a combination of open specifications and standards. This is so that different groups can develop these individual tools and services with confidence that they will interoperate.

# 3.e Onramps to the Open Metaverse

We need more than good infrastructure for a decentralized and open metaverse to succeed. Users need intuitive interfaces by which they can discover and pay for new experiences, manage and utilize their digital identities, and trade user-generated content. Creators and developers need simple yet powerful tools to develop, deploy, and distribute their creations with fair monetization. But so far, the walled-garden approach of Web2 has seen monopolies on content distribution and access enrich platform owners while cornering content creators and users into economic models that work against them. The open metaverse we imagine, powered by blockchain and Web3 technology, would instead be defined by standards and interfaces that allow creators to determine their own incentive models while ensuring their creations are accessible and interoperable throughout the ecosystem.

While it's possible to operate on blockchain infrastructure directly from the API, the last decade has seen the emergence of a number of tools that help users access shared blockchain infrastructure. Many of these same tools will be needed in the open metaverse, though they will likely take on new forms to fit the needs of a different class and motivation of users. In particular, unified platforms that combine these tools into an intuitive user experience are likely needed to make mainstream use feasible, reducing decision fatigue and the learning curve to get started by limiting the number of applications new users need to adopt. However, unlike the walled gardens that have emerged across Web2, forcing users to adopt multiple gaming, video, and audio streaming apps to access their content, the open metaverse infrastructure should be agnostic to the tools used to access it. While popular tools and even official standards may emerge, anyone is free to develop their own interoperable tools against the same open API.

## Wallets

Wallet apps are one of the most critical Web3 tools a user will adopt, as they are the means by which most individuals will manage the private keys which control their digital assets and in some cases act to authenticate the user. The term "wallet" is a nod to the financial roots of blockchain, and it may be beneficial to move away from it in the context of the metaverse. However, to avoid confusion, we will continue to use it here.

To achieve mass adoption of the open metaverse, these wallets need to be user-friendly and accessible to metaverse citizens with varying levels of technical expertise. They should present clear and concise information at a glance as might be sufficient for the majority of use cases while providing a quick means by which to access more detailed information. They should be compatible with the various assets, applications, experiences, and tools of the open metaverse; capable of transacting with all of its various assets.

Perhaps unsurprisingly, wallets are where many prospective adopters fall off the blockchain adoption funnel. Despite dramatic improvements to user experience in recent years, we have a long way to go before wallets are at parity with Web2 experience in terms of usability. Wallet key generation is a nightmare for the average person who already has poor password practices and when things go wrong there is no help desk available to "reset the password." For wider acceptance and use in the metaverse, wallets will need ways to address these accidents to alleviate the risk that users, application developers, and IP holders face by using blockchain infrastructure. Popular options here include custodial and semi-custodial key management as well as social recovery mechanisms which allow a combination of people you designate to recreate your keys. Each of these schemes introduces its own sets of risks and highlights the opportunity for multiple wallet providers to exist, as we see today.

# Explorers

Explorers are tools that aggregate data from a blockchain into a database so that it can be presented in intuitive ways for users to verify transactions and gain insights into the state of the platform. Traditionally, their role has been one of transparency, allowing anyone to see and verify any transaction on the platform and, in some cases, who is processing the blocks. Likewise, explorers in the context of the metaverse provide transparency and visibility into the metaverse ecosystem. Users can view the history of a digital asset, including its ownership and transaction history. In some cases this might be helpful in the context of seeing how their assets have been used across the metaverse; in other cases, it could help determine the authenticity of assets.

# Marketplaces and Discovery

Discovery platforms will provide a crucial role in helping users navigate what could quickly become an overwhelming flood of content. One hope of freeing creator economics, in combination with the emerging ability for artists to scale their productivity with more democratized tools, is that we could see a renaissance of sorts where people are empowered to explore their creative potential at a scale previously limited to heavily-capitalized organizations.

We've already seen artists become disillusioned with current NFT marketplace solutions and their lack of royalty protections or discovery mechanisms, making the economic viability of participating questionable at best. Metaverse marketplaces will need discovery mechanisms that help users discover content that is relevant based on their interests, current assets, or even social graph and should balance personalized recommendations with diversity to ensure that users are exposed to a range of experiences and digital assets.

# 04 TECHNOLOGY FOUNDATIONS

A lot goes into powering shared virtual worlds. They are complex systems that manage the state of 3D avatars, environments, and objects, i.e. the people, places and things that comprise the metaverse. They have to deliver vast amounts of rich content to client systems with disparate capabilities; they need to render graphics at real-time frame rates; and they must communicate changes between clients so that participants have a consistent shared experience. All this, and they must be connected to open and secure payment systems so that creators and infrastructure providers can be properly rewarded and maintain control over their destinies.

Our approach to providing Metaverse-as-a-Service combines LAMINA1 blockchain technology and future EVM enhancements with best-in-class network services and subnet infrastructure for scaling and optimization. We also intend to offer features that creators and users are clearly in need of, starting with a reliable data storage layer and secure messaging.

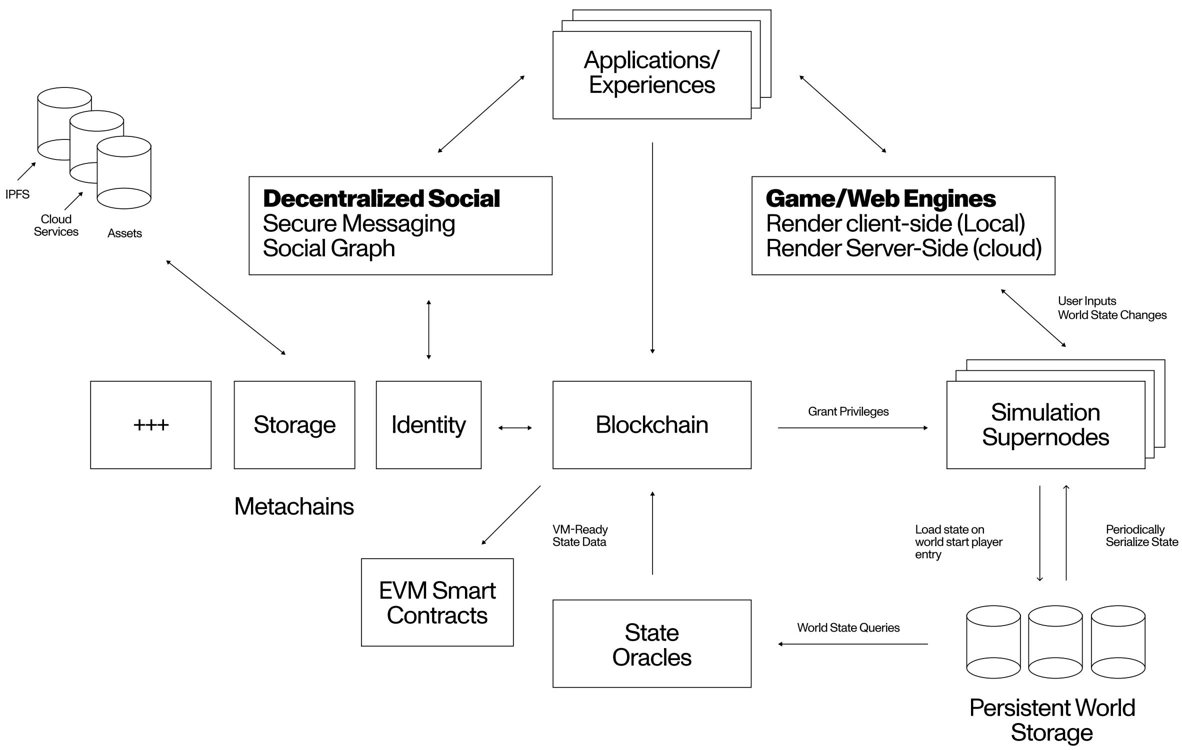
Additionally, there are several areas that LAMINA1 is in the early stages of investigating, including EVM enhancements for 3D computation and spatial reasoning; open and flexible avatar creation tied to open identity; cloud/hybrid rendering; AI-based asset generation; mapping, search and discovery; and decentralized social networks. All of these will be essential components of the open metaverse over time, though they are not our initial focus areas.

In the future, such solutions may be LAMINA1-sanctioned or approved MaaS services to ensure quality levels of service and user experience — though in the spirit of Web3, we would like to foster a completely open ecosystem where anyone can be a Metaverse Service Provider by adhering to upcoming protocol/API specifications and standards.

Admittedly, this is an enormous problem space to tackle — beyond the capabilities of any one company, or even consortium. Our intent in this whitepaper is to frame the issues and offer a handful of starter solutions. We are also eager to work with like-minded stakeholders to create interoperable pieces in a bottoms-up fashion—though driven by envisioning specific outcomes in the open metaverse.

# 4.a An Architecture for the Open Metaverse

Any architecture for the open metaverse should strive to bring together the open nature of the web with the power of real-time 3D virtual worlds and web3 foundation. Many existing web services will just work out of the box, but some innovation is required, such as new media formats (or extensions thereof) and new protocols for multiplayer messaging. It also must be designed with blockchains and decentralization in mind and should be future-proof for a range of devices, from mobile phones to fully immersive headsets (though not be tied to any of them).  Many organizations will be working on standards and protocols, and our proposed architecture assumes those standards will fall into place over time. Until they do, we are offering our initial thoughts on the pieces that will be required.




*MaaS System Architecture*

The above diagram depicts our vision for Metaverse-as-a-Service as one potential system architecture. In the following sections, we will describe a combination of blockchain technology, dedicated domain-specific subnets (i.e. *metachains*, see below), and time-tested distributed simulation/gaming technology.

All will work together to form the foundation of an open, interoperable, secure, and reliable network where creators can build a broad range of platform-independent, immersive multi-user metaverse experiences.

# 4.b Metachains for Flexibility and Scalability

Historically, a lot of energy has been invested in funneling blockchain developers onto a small group of blockchains, the majority of which have a single shared EVM (Ethereum Virtual Machine) environment in which all users, apps, and smart contracts compete for resources. The result for many developers has been limited transaction throughput, high latencies, and unaffordable or unpredictable pricing. There are technical and economic reasons for this. While reliance on a single blockchain was viable early on, where successful applications measured their active users in the thousands, mainstream adoption requires a different approach.

We believe that to support real user volume, some or most metaverse experiences are going to need their own dedicated blockchain infrastructure. The term "appchain" has emerged to describe a blockchain that is designed to serve the needs of a specific application. This approach attempts to address some of the scalability, performance, and economic issues of general-purpose blockchain networks in a number of ways:

- The consensus mechanism can be tailored to a single application, ensuring performance and fees for a specific application are not heavily influenced by the usage of unrelated applications running within the same VM.
- Application-specific chains can reduce competition for network resources, such as EVM, bandwidth, and storage, resulting in significant performance gains.
- The economics and governance models of the network can be suited to fit the varying needs of metaverse application developers. For example, larger creators are important to a compelling metaverse as they can bring high-production-value experiences which galvanize energy and adoption but they generally have a higher sensitivity to regulatory and brand risk and need more control over their infrastructure in order to enter the space.

As with most things, this approach does have its drawbacks. Launching a standalone blockchain can be a complex process that requires relatively deep knowledge and expertise not only on the technical side of customizing and deploying the chain, but also in building out a community of validators and users, designing economics and governance, and integrating with an ecosystem of tools (e.g. wallets, explorers, indexers, RPC providers, and exchanges). Additionally, providing interoperability between these dedicated blockchains can mean relying on centralized entities to bridge your chain with others, introducing real security risks for users.

To address these challenges, LAMINA1 is committed to supporting the easy deployment of dedicated decentralized infrastructure for metaverse experiences using the subnet architecture developed by Avalanche. These specialized subnets, or **metachains**, will form foundational components of the LAMINA1 blockchain, allowing creators to specify the economic and governance properties of their subnet to meet their application's needs while maintaining a measure of interoperability with the rest of the LAMINA1 ecosystem.

Creators can also deploy custom virtual machines as metachains alongside their blockchain (or other data structure) which define the types and ways that transactions are processed within their subnet. While using a standard Ethereum Virtual Machine within a dedicated metachain already provides creators flexibility and performance gains, we plan to support and develop a number of custom virtual machines that are tailored specifically to different types of metaverse experiences, providing much higher performance at the sake of some composability on the chain after it's deployed.

Metachains allow the underlying infrastructure of the metaverse to grow on demand, scaling to support many thousands of interoperable applications and experiences, each capable of supporting real mainstream adoption through the use of custom VMs.

# 4.c Blockchains for Virtual Rights Management

The LAMINA1 blockchain is being developed specifically to support the needs of the metaverse, i.e. rich interactive media, real-time 3D graphics, and multi-user virtual worlds. While there is a temptation to put all business rules and governing logic for the metaverse "on-chain" to foster decentralization and transparency, it is just not viable given the state of blockchain technology today.

For example, blockchains are not appropriate for use in real-time simulations, which require 30, 60, and even potentially 120 updates per second shared among multiple participants. They are also not well-suited for serializing all the data for a persistent world, such as storing the positions of objects in a scene after they have been moved — classic database Create, Read, Update, and Delete (CRUD) operations are more appropriate for operations like that. And finally, while great strides have been made in blockchain programmability via the Ethereum Virtual Machine (EVM), the EVM's capabilities are nowhere up to the task of supporting the 3D computations required to support many required operations for functioning virtual worlds.

This suggests that we focus our attention on what blockchains do well: They are distributed, immutable transaction histories, with some limited programmability, governed by consensus algorithms intended to be beyond centralized control. Blockchains are already proven for payments — now let's look at how they can also be used to safeguard permissions in decentralized virtual world systems.

## Virtual Rights

The idea of "digital ownership" is quite complex. "Ownership" itself is actually a composite of more basic rights that we should unpack and understand, especially since the full set of rights are not always transferred, as expected when purchasing a digital thing (e,.g, purchasing an NFT might not convey any copyrights).

In reality, we have an array of legal rights, such as the right to use, possess, transport, control, sell, loan, derive from, and often (but not always) the right to change or destroy a thing we might think that we own. Copyright and common law give us related rights and responsibilities. The fuller set includes the right to loan, view, copy, republish, mash-up, enter (a space or contract), or even to critique or lambast something (such as "fair use" allows for even copyright-registered works). These rights could apply to objects, clothing, vehicles, land, buildings — pretty much any digital objects in the metaverse. They can also be shared via co-ownership.

For simplicity, we will start with a common language for talking about these rights.

# Table of Common Rights Terminology

| Common Actions | And the Corresponding Right to... |
|---|---|
| Obtain / Manipulate/ Release | Receive, move, scale, rotate, track an object |
| Create / Derive | Create objects & spaces (ideally with authorship tracking as you might see in GitHub |
| Modify / Destroy | Change objects or spaces, considering dependencies |
| Enter / Exit | Go into or leave a given space |
| Portal In / Out | Create portals in and out of a space |
| Annotate / Republish | Reference or augment separately (as with syndication, comments, criticism, etc...) |
| Elect / Manage | For a supernode managing a given space in time |
| Transfer / Sell / Loan / Assign | Assign certain (up to all) rights to others, for a given period of time or permanently |

Unlike the real-time portion of a running simulation, the permanent record on the blockchain only needs to keep track of who granted, sold, or leased which specific rights to whom, and which servers are authoritative for those things during any given period of time (see discussion of *supernodes* below).

Users may implicitly exchange some rights during normal activities. For example, Alice and Bob are Metaverse participants in the same virtual place and time. Alice hands a virtual item to Bob, say a framed photo. In this case, no ownership change is implied, and no exchange of funds happens, but Bob has current possession of the object and can hand it back, throw it away, or attempt to retain it indefinitely. Importantly, the exchange of rights happens first in the real-time simulation but is later reflected with authority in the permanent record, should any disputes arise.

An important consequence of these transactions is that everything is consistently time-coded, including real-time events, as well as the beginning or end of a given period of rights holding. This allows the real-time portion to never have to wait for the permanent record to update itself, as it can use the last-known-good state and project that until some anticipated endpoint is reached. In fact, efficient transfer of rights should always be established "as of" a start time and "until" some end time in the near future, so everyone has a chance to align with the changes before they happen. As in the case of a ball bouncing around a racquetball court with players eagerly trying to hit it, it's not always possible to predict too far ahead.

The concept of "ownership" often includes the right to modify or destroy something or to otherwise transfer or assign some or all rights. But any given user can only assign the rights they have. If Alice again gives the ball to Bob, there is no implicit right for Bob to destroy it, though that may happen as a result. However, Alice can reclaim it from anyone who has possession by demonstrating her set of "ownership" rights.

There has been interesting work in rights management within the Web3 and metaverse communities to date. Notably, The Sandbox and Decentraland innovated on virtual land models for the Ethereum blockchain in the form of 2D subdivided parcels for sale. However, those models are not broadly generalizable to higher-dimension spaces, objects, avatars, and open markets. Some Web3 developers are offering token-gated access (such as for online concerts) as a form of conferring rights to specific experiences or websites. But for these, most of the rights listed above, other than the "right to enter," are not coded in such a way that they can be handled in-world or on-chain. There needs to be significantly more work done in terms of handling the nuances of different rights, collisions of rights, transferability, existing laws and regulations, and the safety and privacy of rights-holders.

# 4.d Hybrid Networking for State Synchronization

While blockchains are ideal for a permanent, authoritative data store with some programmability, even the fastest ones aren't fast enough to keep a running 3D simulation up to date. State changes to the virtual world, such as avatar and camera movements, object animations, and user interaction, must maintain real-time game frame rates to deliver a satisfying user experience, updating the simulation 60, 90, even 120 times (frames) per second.

In a shared virtual environment, such changes not only need to happen rapidly, but any updates to aspects of the environment that would be visible to other users also need to be communicated to all the other clients running the simulation. In addition, changes must be communicated in a way that is trusted among all clients, i.e. some entity or entities are designated as arbiters of the authoritative state of the world. This is a well-understood problem — distributed simulation — with a long history of development going back to the 1980s for military simulation applications. Since then, it has been refined at scale in consumer applications for more than three decades with massive multiplayer gaming.

There are many existing solutions for multiplayer networking that could provide a base layer of functionality for Metaverse-as-a-Service. Note that one size does not fit all: Peer-to-peer networking can perform well for an 8- to 15-person chat space but is inadequate once a group size reaches several dozens to hundreds of users — the point at which most systems rely on centralized server mechanisms for authority and efficient communication. Even within central-server models, interesting tradeoffs must be made to achieve mass scale with performance: different systems employ techniques such as sharding, spatial partitioning, area of interest management, and potential visible sets, but each of these techniques of necessity introduces limitations to functionality.

For example, *sharding* is a reliable way to divide millions of players into multiple independent instances of the same game, each with only hundreds of players. However, those instances don't talk to each other or share state, and their users might as well all reside in different universes. On the other hand, sharding could be a good method for supporting completely independent groups of users communicating in different versions of the same environment — e.g. you and I meeting together at the top of a virtual Eiffel Tower, where we want to communicate with each other but don't actually want other people overhearing our conversations.

In the earliest military **Distributed Interactive Simulation (DIS)** systems, nodes were highly decentralized but suffered from a persistent problem, most easily visualized as two opposing tanks (call them "A" and "B") simulating a battle across two or more computer systems.

For example, imagine if tank A thinks it kills tank B and, at roughly the same time, B thinks it kills A, with the relevant network messages passing each other in transit. If A disabled B first, then B couldn't have shot A. The same could be true in reverse. Such is the price of network latency. In this situation, each system would need to check the message time stamps, synchronized to a common clock. And if A really hit B first, then the computer supporting tank B might need to "undo" their short-lived victory (aka "rollback" state), which can be confusing, to say the least, like a kind of anti-deja-vu.

Later consumer systems, like **Second Life**, tried solving this with highly centralized authority to answer those tricky precedence questions — physical simulation and handoff of objects — in a single server per plot of land (more on that below). They kept remote clients as very thin shells, mostly for rendering. But that approach failed to scale, because the user input always happens at the client, forcing a long round-trip to do anything. And servers can only handle so many concurrent participants, roughly hundreds each. **MMOs** relaxed the requirement that every player sees the same exact world, and thus sharding allowed 100 instances of 100 players each, to handle 10,000 simultaneous experiences. But as noted, sharding is not appropriate for all use cases.

The next set of ideas were **adaptive hybrids.** In small groups, say eight or less, peer-to-peer client topologies are more efficient than sending everything to a centralized authority. Above a certain number, the cost of each client sending a unique message to each other client becomes more expensive. In any event, each client can simulate its own world, but then vote on the shared truth when they disagree. If this is coupled with more robust prediction, such that the simulation can safely look farther into the future, the system can avoid rollback almost entirely.

For example, once the shell is fired, we know where it's going to land. The only question is if the target moved quickly enough to get out of the way. So any corrections can be much more subtle, based on the last millisecond user inputs that physics can't predict. This approach is more common in games with limited numbers of players, say eight on a LAN.

In larger groups, peer-to-peer becomes more expensive than going through a central router, and we have to adaptively add a shared repeater at the very least. This need not be "centralized" in the sense of only one provider — it could be adaptive as well — as long as the current participants agree on a single router at any given time.

**Croquet*** is a prime example of that approach, where all the simulation work is done in distributed fashion but messages can be efficiently routed. They indeed reflect all dynamic user-input signals through a centralized relay. But the shared reality is consistently and independently simulated and predicted on each client, independently, as a result of any user changes. That means millions of moving objects don't need to be synchronized because the simulations will be identical. The only external unknowns are the choices made by humans to change the simulation similarly across all clients.

**Photon*** is another commonly used networking library, especially for games made with Unity. It offers its own cloud resources to help scale games to tens of thousands of concurrent players. But it is still centralized. Basic networking is included with Unity for simpler use cases.

*https://croquet.io/
*https://www.photonengine.com/#

None of these solutions are perfect, shifting bottlenecks and compute costs from servers to clients and back. But breaking the problems up into smaller chunks allows the greatest flexibility in adapting to meet high-performance demands. Adaptability is key, especially given the dynamic number of participants in any given space.

To provide the most flexible set of choices for creators, LAMINA1 will encourage ecosystem developers and prospective partners to provide specialized synchronization services that address the whole range of use cases. For example, a classical DIS-style network of clients may consist of totally peer-to-peer connections, or later from a concept of supernodes acting as temporary authorities and activity hubs. Croquet.io has a different approach, also worth supporting here, where it sends only the user-driven state changes since each client is running a bit-identical deterministic simulation that doesn't otherwise need re-synchronization, except for user-driven inputs. Even random numbers are identical across clients.

Given the variety of use cases and solutions for synchronization, it doesn't make sense to mandate that servers exchange messages indicating the latest position of all objects as they change. The only requirement we can ask is that metaverse service providers working within this proposed ecosystem handle all real-time state synchronization themselves but work with the proposed rights-based system for the deeper and more persistent transactions described in the previous section.

Companies are of course free to write their own implementations of virtual rights management too, using metachains or other blockchains or anything else that works. But then they miss out on the cross-platform features we are envisioning for MaaS. To the extent that customers want these enumerated rights across all worlds, something has to be a unifying layer.

For example, if a company deploying Virtual World Service "A" uses a peer-to-peer architecture to send telemetry for all moving objects among all client nodes, they can still use MaaS metachains to store the answers to who has which rights and possessions at any given time. Service A must be able to consistently interpret the set of granted rights and times. If someone wants to make a portal from a spot in Service A to a spot in Service B, which let's say was built on Croquet instead, then those two services may have trouble sharing telemetry messages, since Croquet doesn't use those kinds of routine telemetry messages, except for user-driven changes.

But assuming both A & B use MaaS rights management, they can exchange consistent state information about who has which rights and how the portals should behave, such that two people could hopefully play catch across a portal. Hopefully, all such services will enjoy common content-sharing standards (being developed in partnership among many companies) to render these portals in a rich way. But solving the portal problem itself or the standards that enable it is beyond the scope of this whitepaper.

# Supernodes, Simulation Authority and State Oracles

Given today's reality of serializing data to blockchains in real-time, we need to turn to more time-tested means of governing the running simulation, e.g. deciding who or what is "in the same place at the same time."

As multiple users come together at some virtual location, we need a commonly agreed authority on what officially happened during their interactions. And when it is time to periodically serialize that state so that users coming into a running simulation experience the most up-to-date version, we need a similar type of authority governing those operations to ensure security and maintain trust in the outcomes. These dual authorities could be part of the same running system, and given the need for tight coupling between these services, likely would be in practice.

To establish these authorities, we propose a scheme similar to virtual rights management above: users dynamically elect a supernode from among them to exchange their real-time transactions and handle serialization, where the more slowly-changing scene and world properties are saved to subnets. As with other virtual rights, the supernodes' authority would be time-boxed. The supernodes and their durations would be reflected in the permanent record on-chain to resolve most disputes that may arise.

The real-time portion of the metaverse needs to run in microseconds per transaction, tracking the location of users and objects in space to answer questions like: Where is it? Did it collide with something? Where is it going? We use physical simulation and prediction to answer many of these questions from the recent past and into the near future. Cloud resources could definitely do this work for some number of users, as with Second Life or newer back-end simulation services. But that forms another kind of centralization bottleneck and cost. We can do better.

Alternatively, we could dynamically elect one of the running simulation servers to handle the authoritative state management for each small group. We can even change the election over time, rotating responsibility to keep things fair. Each nearby user connects to the supernode covering the small region they're in. This relays interactions from other nearby users, and to some manageable extent, the greater virtual world around them.

The unique address of the supernode is stored on the permanent record for unambiguous routing. Each supernode is itself responsible for remembering what happened within its own limited domain in space and time. In the event that a supernode is asked to retain a certain unusual set of interactions (say: a record of a bad actor or user abuse) for a longer period, it will make a protected copy and offload a reference to blockchain storage (see below) for more persistent storage. In general, supernodes only need to keep track of the time period for which they're elected and retain this information for only a short period of time after.

From a blockchain point of view, if any computations or logic require referring to a running simulation's state, smart contract code can consult blockchain data as written by external processes that interact with simulation supernodes. In blockchain parlance, we call these processes state oracles. State oracles would periodically write VM-ready data to the blockchain so that EVM or other virtual machine code can consult the blockchain for answers about the current world state, for example, to determine if a world has one or more players currently inhabiting it. As a measure against tampering, state oracles should also record the ids of supernodes, time-indexed, to allow for validating state queries against the supernodes that were duly authorized during the time period.

# 4.e World State Persistence

Related to real-time state synchronization, there is a need to keep virtual worlds alive, ongoing, and consistent regardless of how many people participate at any given time. We call this world state persistence. It is the answer to the question: “What happens when the last participant leaves a space and someone rejoins a week later?” (i.e., if a tree falls in the forest and no one is there to watch it, is it still fallen vs. reset when we return?)

In this case, there would be no consistent supernodes to remember the live simulation state. Any cloud resources for services like Photon may have been reclaimed for other active uses after just a few minutes of inactivity. Croquet solves this for its spaces by collecting up a large bundle of state changes (a snapshot) that any newly joining participant (i.e., their client running Croquet) can apply in sequence, plus any last-minute updates since the snapshot was taken. This propels any new participant into the current synchronous world state. Photon may solve this differently, by sending a current list of all dynamic elements to any new joiners. Some of these approaches are highly centralized, though one can imagine future federated versions along the lines of what we are envisioning for decentralized asset storage (see next section).

What a MaaS architecture expects from any participating network system is the ability to serialize world-state to some semi-permanent form, possibly cloud storage, where only the URL and metadata of this bundle is codified on the metachains or on the main blockchain for any person who later re-joins this space.

We can define a small number of actions to handle these needs:

| Prototype | Return on Success | Meaning |
|---|---|---|
| Save Space (superID, spaceID, agentID, endT) | URL | Command to world simulation service to serialize state in a way it can unpack later. |
| LoadSpace (superID, spaceID, agentID, beginT) | spaceID | Recover a previous world state to resume live simulation |
| DeleteSpace (spaceID, agentID, beginT, endT) | True or Error | Delete storage for any snapshots between begin and end times. |

These protocols need a *spaceID* to point to the unique space to be preserved or recalled. The *beginT* and *endT* times are used to find the right versions of worlds for a given need. Usually, we want the latest, but not always. For our purposes, all IDs are GUIDs, and Times are measured in double-precision seconds since 1/1/2001.

These methods also need a *superID* and *agentID* to validate the action, as not just anyone can save and load state. An agent that is hosting a supernode has the right to serialize the state of this supernode. Another agent could ask to be approved (via the supernode vote) to serialize state in their place. An agent that has permission to enter a given space should have sufficient permission to recall and rebuild the space, as long as no other instantiations are live (if so, there should be another vote among the participants).

This semi-persistent form of world state is not meant to be stored forever but is typically stored until securely replaced by another more recent update. However, keeping multiple versions of data around is also useful for debugging or investigating bad behavior days, weeks, or months after the fact. These temporary storages should never be depended on as long-term storage, since they each may be replaced by newer versions.

Given the typical tight coupling between simulation servers and world state storage, it is reasonable to expect that delegated supernodes would also be considered the definitive authority for storing world state, and that delegation information should be stored on-chain.

# 4.f Reliable Decentralized Asset Storage

One of the more interesting aspects of the intersection of Web3 and the metaverse is the agency granted to users to own and control their own assets and potentially bring them across metaverse worlds and experiences. A lot of energy has been devoted already in the field to the development of non-fungible tokens (NFTs) to represent various assets of an immersive experience, including characters, avatars, game items, land, music, and more. By holding one of these tokens, users can have irrevocable ownership and control over these tokens within the bounds of the smart contract that created it. This enables all kinds of interesting ways to interact with and use them including: selling them in marketplaces, loaning them to other players, bringing them into other experiences or games, upgrading them, etc. LAMINA1 intends to build on this body of work by tackling some of the bigger infrastructure needs behind it — namely the reliable decentralized asset storage required to empower seamless, fast, and interoperable metaverse experiences.

## On-chain vs. Off-chain Asset Storage

A significant hurdle in creating and using these NFTs is the sheer amount of data (and in turn, cost) that would be required to store each asset completely on-chain. To address this, NFT data is usually separated into an on-chain component and an off-chain component, with the goal of limiting on-chain data to that which is required for any computational interaction with the asset on-chain.

For example, imagine a decentralized trading card game where the core logic of the game is managed on-chain using smart contracts rather than using a centralized server, while rendering is done client-side on the player's computer or phone. There are aspects of each in-game asset that the game logic smart contract needs to run the game, such as: the asset name, type, stats, experience, reputation, rarity, owner, etc. Traditionally, this is all information that needs to be stored on-chain for the smart contracts to access — but there is a great deal of additional asset data (usually in the form of graphical data) that is really only needed by off-chain clients such as the local rendering engine of the player or a marketplace that is selling the asset. This information is usually stored off-chain in either a decentralized storage system like IPFS or a traditional centralized server like an S3 bucket, with a link to the data stored on-chain within the NFT's metadata.

Breaking up asset data like this, while relatively cost-effective, introduces some obvious potential issues. First, if the storage system for linked off-chain data isn't reliable or gets shut down for whatever reason, the NFT can become unusable in its intended environments. One infamous example of this problem was the loss of millions of dollars of Coachella NFTs* (along with countless other digital assets) that were stored in a centralized manner within FTX. Additionally, most blockchain virtual machines are unable to perform computations on off-chain data, making it difficult or impossible for the VM to meaningfully update, manipulate, or render the data stored off-chain. Finally, given our focus on facilitating immersive experiences, it's critical that any off-chain storage needed by the experience has very low latency, high throughput, and high availability.

*https://decrypt.co/114856/coachella-tomorrowland-solana-nfts-stuck-ftx

# Goals

With all of this in mind, LAMINA1 has identified a number of requirements for non-centralized asset storage in the open metaverse:

- Must provide the ability to store, update, and retrieve potential petabytes of data per experience
- Must have extremely low latency (<0.5 sec) loading assets dynamically while in-experience
- Must have relatively low latency (<5 sec) while outside of, entering, or transitioning between experiences
- Must be interoperable across open metaverse experiences
- Must offer high reliability (no significant downtime)
- Must offer high confidence in data persistence
- Must provide asset owners the ability to control access to their data and assets
- Must be secure against malicious actors
- Must offer an intuitive, predictable, and cost-effective price structure and business model

A lot of great work has been done by other projects such as Filecoin in building systems that get us close to these goals. Rather than reinvent the wheel on our own, our focus is to collaborate with these teams to optimize their technology for open metaverse experiences. We have several ideas in mind that would accomplish this using dedicated subnets, or metachains.

For example, we could facilitate subnets that provide, as a service, persistent storage of assets using existing protocols. These subnets would store and serve content identifiable data (CIDs) for any experience built on LAMINA1. We envision that there could be many such subnets, each run by different groups and with different properties, some of which may even link to existing independent third-party decentralized storage networks. Asset developers can choose which subnets to use based on their requirements or build redundancies by storing across multiple subnets.

We could also couple dedicated file storage within the subnet virtual machines that we offer to metaverse creators. This file storage would serve as a sort of high-performance cache, replicating the off-chain data of all personal assets of a user as they enter the experience, and releasing them when the user exits the experience. Depending on the performance requirements of each metaverse experience, developers may choose to have a fully permissionless validator set within their subnet for stronger decentralization with less predictable performance, or a permissioned validator set that only allows demonstrably high-performance nodes to validate.

Eventually, with a dedicated storage cache within the subnet VM, it could be possible to access asset storage directly from smart contracts or other computational environments executing within the same VM. This could open the door to more interesting use cases like computing across large data sets, remote rendering, and managing complex world states in a decentralized manner.

# 4.g Secure Decentralized Messaging

LAMINA1 believes that facilitating online interaction across the metaverse and internet at large in a safe, open, and decentralized manner is foundational to our vision and should be a core function of Metaverse-as-a-Service.

There are countless messaging and social platforms already, but the majority of them are centralized and operate in walled gardens, preventing adoption and deep integration with immersive experiences.

The communication protocol in an open metaverse should have at least the following properties:

**Decentralized**
The infrastructure for communication in the metaverse should not be controlled or gate-kept by centralized entities. Any user should be allowed to run or choose their own infrastructure and communicate across the platform, and all code should be open-sourced.

**Private**
At a minimum, messages going across a decentralized network should be end-to-end encrypted, otherwise anyone in the network can read them. An additional step would be to make them metadata resistant, which would keep anyone in the network from being able to build a social graph, knowing who all your contacts are without your consent.

**Interoperable**
Any application should be able to integrate or build a client according to a set of standards or API that gives their users access to a communications layer that spans across many different metaverse applications and experiences.

**Performant**
The communications infrastructure should be able to support millions if not billions of users with end-to-end latencies of less than 3 seconds. Performance within an experience might have much higher performance requirements to enable real-time coordination within the experience while communication across experiences might allow for longer latencies.

**Moderatable**
A means by which to moderate public spaces is required to prevent abuse and spam of the platform.

Global events have driven a lot of energy toward solving this problem and we plan to build on that work for MaaS. There are a number of approaches that have been developed to provide a more open and secure social infrastructure on top of the internet:

### Peer-to-peer (P2P)

In this approach, messages and content are sent directly between devices without going through a central server. This makes it difficult for third parties to intercept or monitor messages, but it also requires both parties to be online at the same time. Examples of this approach include: **Status***.

### On-chain

This approach uses blockchain technology to secure and store messages and content and ensure their authenticity. Content is stored on the blockchain, which means it cannot be easily altered or deleted once they have been sent but are easy to retrieve. This approach is highly secure, but it can be slow and resource-intensive. Examples of this approach include: **DeSo***.

### Federated

In federated social, decentralized servers communicate with each other to enable messaging between users. In this approach, users trust or rely on at least one server operator to send and receive messages, though they generally are able to communicate with users on other servers. This approach provides a decentralized architecture, but it still relies on trusted server operators. Examples of this approach include: **Matrix***, **ActivityPub*** and **Mastodon***.

**Decentralized Off-Chain:** This approach leverages a decentralized blockchain network to manage and coordinate a network of decentralized servers. Similar to federated social, these decentralized servers communicate with each other to enable messaging between users but they distribute any trust assumptions across multiple servers in the network improving the availability of the network and providing redundancy in the case of server downtime. Examples of this approach include: **xx Network***, **Nym***, and **Session***.

*https://our.status.im/status-launches-private-peer-to-peer-messaging-protocol/
*https://www.deso.com/
*https://www.matrix.io/
*https://activitypub.rocks/
*https://mastodon.social/explore
*https://xx.network/
*https://nymtech.net/
*https://getsession.org/how-session-protects-your-anonymity-with-blockchain-and-crypto

Each of these approaches has its pros, cons, and individual challenges and we are considering each of the example networks and protocols listed as potential solutions for providing the social layer of the LAMINA1 metaverse. Some of the considerations we're weighing in analyzing these options include: Message encryption, message transmission, message retrieval, interoperability, monetization, and rich social functionality.

Message encryption is the means by which message content is kept private and only accessible by the sender and recipient of the message, obviously an important property in an open network where anyone can intercept traffic. We need a robust key establishment and management protocol that ensures two users can create and maintain a secure communications channel that's performant, can be performed asynchronously, and is resistant to censorship or man-in-the-middle attacks. We value a protocol that includes forward secrecy to ensure that if keys are leaked or revealed, only a limited amount of encrypted data is affected. Key re-establishment should be similarly performant and robust, especially given the decentralized nature of the network and lack of centralized key management orchestration.

Once a message is encrypted we consider how it traverses the network and what that network looks like. Does the protocol rely on a dedicated network of servers to transmit the message? If so, what expectations on uptime and availability are there? How much throughput can the network handle? How is integrity maintained and verified as the message traverses the network? Is the metadata of the data packets obfuscated, preventing anyone monitoring the network from building social graphs on the users?

A particular challenge in decentralizing social communication is message or content retrieval. Unless both the sender and the recipient happen to be online at the same time, the content needs to be stored somewhere until the recipient comes online. If content storage is on-chain then we need an optimized virtual machine for handling the massive amounts of storage and transaction throughput that a widely used social platform would require. Additionally, we would need some way to obfuscate or remove messages or content from the chain to respect users' privacy in case encryption keys are somehow leaked. If content is not stored on-chain then we need an off-chain storage solution that is decentralized and has strong availability and redundancy properties. This could be a dedicated set of servers which is either federated or managed by a blockchain.

Monetization of the communications layer is another major consideration. Beginning with email and now with messaging and social media, we've become accustomed to our means of communication being free, usually by giving up our data in exchange. It is a very steep hill to climb to get users to begin paying for messaging so it likely needs to either be extremely affordable or free, perhaps through feature bundling, subsidization, or a subscription-like model.

The social layer needs to be interoperable, allowing communication not only within a particular application or subnet but potentially across all applications built on LAMINA1. This is achieved through defined communication standards and APIs and an underlying network of clients and servers that use them. These same standards should also account for moderation and rich social functionality like reputation, identity, user discovery, and content discovery. **Matrix** and **ActivityPub** have done an immense amount of work towards these standards for messaging and social media in a federated context.

Regardless of the chosen architectural approach, LAMINA1 will likely adopt an active and existing set of standards or partner with a project that has already implemented a solution that can be customized to work with our subnet architecture and which has sample clients that our developers can easily use to integrate rich social into their creations as part of the MaaS package.

# 4.h Relationship to Game Engines and World-Building Tools

At this juncture, it is important to point out what Metaverse-as-a-Service is **not** — namely, that it is not a toolset. Instead, think of it as a collection of system services. Tools are used to generate content and deploy to live MaaS services, but they are not part of the core MaaS effort. That said, without high-quality enabling tools, MaaS will be of limited value.

Low-code and no-code tools will be essential for the long tail of builders in the Metaverse. A creator could start with templates or remix/mashup existing experiences; or import content from modeling and animation packages, import other media, and the right thing happens. This type of workflow will eventually comprise the lion's share of Metaverse content creation.

More sophisticated experiences will be made by programmers and technical artists using game engines and libraries that connect to MaaS services.  Unity and Epic Games both offer full suites of multiplayer and collaboration services that could potentially be used in a Metaverse-as-a-Service context — though neither of these companies has released products specifically built for the Open Metaverse to date. At the high end, authoring solutions such as **NVIDIA Omniverse*** enable the creation of photorealistic digital twins of facilities, cities, and complex CGI productions that potentially reach end consumers via Open Metaverse clients in the form of simplified client applications (with appropriately down-scaled content), or fully photorealistic experiences rendered in the cloud and streamed to less capable devices.

Generative AI art tools such as **Midjourney*** and **DALL-E*** have captured the world's imagination. These tools are rapidly evolving to incorporate 3D creation features, and we are already seeing nascent efforts to generate entire 3D worlds from simple text prompts***. We expect that such tools will become critical to the workflow of artists at every level of sophistication, and further democratize the creation of worlds for the Open Metaverse.

As a starting point for all this, LAMINA1 will provide APIs and SDKs for popular game engines, creation tools, and the web — starting with basic access to capabilities of the Lamina1 blockchain and a growing feature set in subsequent releases. We are also partnering with other blockchain game engine SDK providers such as **OpenMetaDAO / Emergence** to have first-class support for the LAMINA1 blockchain in their tools.

*https://www.midjourney.com/
*https://openai.com/product/dall-e-2
*https://www.youtube.com/watch?v=wU1cQcQFimw
*https://www.engadget.com/nvidia-ai-model-get3d-virtual-worlds-objects-130021127.html
*https://a16z.com/2022/11/17/the-generative-ai-revolution-in-games/

# 05 AREAS FOR FUTURE RESEARCH

Our vision of the open metaverse comprises many technological moving parts, from simulation and rendering, to networking and asset delivery, to blockchains for payments and permissions. In this whitepaper, we have explored several problems that are solvable in the near term through a combination of product innovation, system integration, and standardization. However, other areas of this vast space are so nascent, or not quite market-ready, that it would be better to view them on a longer time horizon as areas for active investigation.

The LAMINA1 core team hopes to have energetic conversations about these topics and build with stakeholders from industry, academia, and standards groups in the coming years to help advance the state of the art.

# 5.a Metachains and Scenegraphs for Spatial Organization and Optimization

As discussed previously, LAMINA1 metachains can provide enhanced performance and scalability by subdividing the blockchain into multiple interoperating chains, each focused on particular problem domains.

Example use cases for metachains include…

**Metachains for Entire Worlds**: Each world operator could run their world on their own metachain, effectively creating a "walled garden," but presumably one with bridges to other worlds in the open metaverse (interoperable tokens, avatars, world content) - thereby getting the best of both situations: an efficient, controlled environment for the best experience, without trapping creators or users in their system.

**Area of interest management**. A running metaverse world can cut down on network load by running "regions" of a large world on a metachain.

**Sharding**. As noted previously, sharding has limited applications. If a metaverse experience is supposed to be a single, common world that anyone can visit, sharding will not work. However, if a sharded experience is appropriate, metachains are an attractive choice for partitioning the database to alleviate network congestion.

**Scenegraphs** are a time-tested tool for managing vast distributed spatial information. Google Earth routinely streams a planet's worth of geospatial information (tera-pixel resolution) down to small web clients and originally to 1999 PCs over 56k baud modems. It does this by managing all this data in a giant but sparse graph, mainly a quadtree with levels of detail, that allows for any missing data to be proxied by lower resolution data until it arrives. In fully 3D worlds, octrees are used to similarly manage scene complexity. The big challenge for Metaverse-as-a-Service will be optimizing this data across many distributed sources.

Scenegraphs can help manage the assets for complex virtual worlds and potentially be decentralized and distributed among multiple data providers. We have working examples of modestly distributed data in the Fediverse, enabling Mastodon and others to share social feeds without Twitter's typical centralization challenges. The main challenge in the *Fediverse* is that people who might like to run their own servers face the same costs and scaling issues as centralized hosts, but without their economies of scale. It's not necessarily cheap, especially with a lot of users involved. Many other systems wind up eventually returning to centralized ownership, as with the largest cloud providers.

# 5.b Spatial Virtual Machines

While there is a strong temptation among blockchain purists to try to put all data and logic on the blockchain, this is not always appropriate. As of today, it is mostly not even feasible for virtual worlds. Objects are complex, have many attributes and much of the data is inherently three-dimensional (e.g. positions and orientations, bounding boxes and volumes). The storage and network constraints of today's blockchains would make 3D storage a time-consuming, gas-guzzling affair.

This situation is even more dire when it comes to the Ethereum Virtual Machine (EVM). Touted as a "world computer," the EVM marries decentralized computation with consensus algorithms to support trustless systems. It is groundbreaking technology that is proving to deliver on its promise within certain domains. However, the EVM is woefully underpowered for 3D computation. Its lack of real number math (fixed or floating point) makes it a non-starter for anything but the most trivial math. What works great for finance does not work well for trigonometry and linear algebra, the foundations of real-time 3D math.

But the prospect of being able to do distributed, trustless reasoning about certain things in a virtual world is too attractive to dismiss out of hand. Being able to answer questions such as "Can this object be placed in this region of space?" based on geometric constraints, in addition to permissions, would mitigate the need for state oracles and other system dependencies (and potential attack vectors). In theory, it could be even more efficient by virtue of being distributed –– if the proper machinery were in place.

It is quite easy to envision a Spatial Virtual Machine, either a new type of virtual machine or an upgrade to the EVM, that supports floating point numbers and the requisite built-in mathematical library functions for trigonometry and matrix computations. A good starting point would be to implement a dedicated high-performance virtual machine subnet that integrates the Web Assembly feature set; this would enable programming on-chain operations in JavaScript or Rust, high-level languages that are quite capable of complex computations with good performance.

**Croquet.io** is a good example of a distributed VM that can run arbitrary JavaScript code. Each client entering a given virtual world runs a bit-identical copy of the world's simulation and rules. Any user-driven changes go through a network repeater which ensures that events happen in the correct order and with consistent time stamps. Any function that might give a different result on different machines is replaced by functions that run the same everywhere. If anyone adds code, that code is replicated among all of the present clients. This obviously needs security to avoid malicious code. But it also represents a good proof point that a JavaScript-based VM with a mutually consistent state is feasible for real-time operations.

# 5.c Open and Flexible Avatar and Identity Services

Identity is a critical component of the open metaverse, defining how users connect with each other and represent themselves both personally and cryptographically across games, apps, and experiences.

There is a natural tendency to focus on avatars as the foundation of user identity in the metaverse. Avatars are, after all, the way we express ourselves visually in a virtual environment, i.e. the “face” that we present to the world around us. There are a few promising projects that enable the interoperability of avatars between different platforms and systems. **ReadyPlayerMe*** offers tools for creating and customizing avatars that can be used across virtual worlds; while it is a proprietary system, it has adoption among hundreds of companies and metaverse worlds. The **VRM** format, based on glTF, is an open standard designed for handling 3D avatar data, body language, and facial expressions. But avatars are just the visual skin of our identity. And while interoperability of avatar content is crucial for the open metaverse, it is by no means the most important or hardest problem to solve.

Identity has many facets and functions depending on the context in which it is expressed or examined. Psychologically, we examine our identity from personal and social perspectives and our digital framework for it should ideally encapsulate characteristics from both. Given the sensitive nature of identity and its associated data, it’s especially important to pair any functionality with cryptographically secure protocols and privacy. Additionally, one of the most important aspects of identity in the metaverse is interoperability, not only within the LAMINA1 ecosystem but ideally with external platforms as well.

LAMINA1 distinguishes between four categories of identity:

**Personal Identity**
In the purest sense, one’s sense of self as a human being

**Sovereign Identity**
The extent to which a nation or nation-state grants a person “identity” in the bureaucratic sense (e.g. a passport or a national ID number)

**Self-Sovereign Identity**
The extent to which a known federating “identity provider” grants you an identity (e.g. a Twitter handle that someone controls, a Facebook ID)

**Self-Sovereign / Permissionless Identity**
The ability of anyone to say “This is an identity” online (e.g. an Ethereum address)

*https://readyplayer.me/

High-level, Lamina1 has plans to support:

**Short Term Goals**
- Self-sovereign/permissionless identity
- Sovereign identity links to self-sovereign addresses
- OpenID connect identity links to self-sovereign addresses

**Long Term Goals**
- Support non-federated self-sovereign identity systems (Stephenson's PURDAHs fall in this category, for instance, and would be very useful to have deployed on Lamina1)
- Support "pure" personal identity features (have Lamina1 be able to know it is interacting directly with a particular person)

There are dozens of projects and organizations working on standards and frameworks for identities controlled by their owners rather than the centralized identity providers of the Web2 era. In some of the protocols being developed by groups like W3C and DIF, users can self-authenticate specific characteristics of their identity via credentials and DIDs (decentralized identifiers) allowing users to prove, for example, that they're 21 without revealing their name, birthday, or any other identifying information about themselves. Other projects like Web3Auth focus on leveraging existing Web2 authentication mechanisms to improve the user experience of generating key pairs that can be used for wallets and self-sovereign authentication. Even large corporations like Microsoft, IBM, and Accenture are developing identity solutions based on user-held credentials.

LAMINA1 will contribute to and adopt these emerging frameworks as a basis to enable streamlined user experiences and richer identity characteristics expressed through avatars, reputation systems, and social graphs. This is an ongoing area of research for the LAMINA1 team. We will be releasing an upcoming litepaper on Identity in June 2023.

# 5.d Artificial Intelligence

Over the last few months, the proliferation of AI tools like DALL-E, Stable Diffusion, Midjourney, and Chat GPT has proved that Artificial Intelligence is well on its way from relatively crude online obscurity to a vivid means of creative expression. Of this advancement, opinions are divided, with many asking: what does the influx of AI mean for creators, builders, and developers in the next online era?

On the one hand, AI will almost certainly power all kinds of future metaverse technologies and experiences, from AI NPCs and live-generated art/environments to much-needed creator tools that streamline and simplify the intricacies of creating the next generation of immersive content.

On the other hand, it's important to ask ourselves: if AI can generate art that looks like the style of a particular artist, or output information that heavily references the work of a particular person or organization — who is harmed, who benefits, and most importantly, who should benefit?

Right now, we are very interested in the idea of "value chains" — a proposed swing of the economic pendulum that acknowledges most of the data used in modern AI networks comes from the work of people — people who are not currently economic participants of these networks. In other words, what if future online ecosystems were able to keep track of all "creators" so that future decisions can be made about their economic inclusion, rights payments, AI human equity, and beyond?

While the challenges would be massive, the benefits would be incalculable: A pathway for providers of complex data sets to be recognized and compensated for their work; a way for artists and builders to generate revenue by interacting with these models; a progression in figuring out how we, as a society want to interact with AI while providing its users a means of real participation in the future metaverse.

LAMINA1 is interested in creating, collaborating, and possibly implementing a way to pay the large lists of people that inform and contribute to the data used for training these models — while empowering creators to use AI tools responsibly while laying the foundation for future open metaverse environments. If you're working on something similar or would like to collaborate on an upcoming whitepaper, let us know.

# 5.e Cloud and Hybrid Rendering

The open metaverse will need to balance two consumer demands that are in dynamic tension: increasing visual fidelity and greater mobility. People want access to the best content from wherever they may be and on whatever device they have in hand — at home, at the office, or on the go. And at any given time, a wall-powered device will always be more powerful (and power-hungry) than a mobile device.

Because of this demand and the potential opportunities, there has been renewed interest in cloud rendering –– technology to deliver high-fidelity real-time 3D renderings from a powerful computer in the cloud to a modestly powered device like a mobile phone. The reality today is that, though it has been around for decades, cloud rendering still does not meet the requirements of many use cases in the metaverse.

Various cloud rendering solutions provide a means to provide higher fidelity 3D graphics without burdening the consumer's own devices except for the necessary 2D video and audio decoding. This is most helpful for customers using low-end devices or those who don't want to buy games, but just try them out or rent them. And it works decently for many games, though not all.

The key idea is that a virtual computer in the cloud will run the same exact game or virtual world that someone else might run locally on a PC or laptop. The game's rendered audio and video is sent to a thin client for display, while game or avatar inputs are streamed back to the cloud as quickly as possible. This is possible due to advances in compression speed on the rendering end and a lot of prediction to cover the added latency.

The challenge is that this approach a) Almost always consumes more bandwidth than a pre-downloaded locally-rendered 3D world, b) Still has relatively high latency for things like VR head-tracking (where every millisecond counts and the speed of light is non-negotiable), and c) The cost structure is either dynamically better or worse than buying your own PC, swinging like a pendulum from year to year.

That last point is worth unpacking, and is most likely why services like **Gaikai***, OnLive*, Google Stadia* and other streaming services ran into apparent business viability concerns. Others are still entering the field, like **Amazon Luna***, **Shadow*** and more. Depending mainly on the current price of GPUs and how much each end customer uses a local or cloud solution, cloud rendering could be a win or a loss. Recall that GPU prices have been extra high for several years, given blockchain "mining," favoring cloud rendering overall. But that's shifting back to normal again, so the pendulum may swing back to decked-out PCs.

*https://www.engadget.com/2012-07-02-sony-buys-gaikai.html
*http://onlive.com/
*https://stadia.google.com/
*https://luna.amazon.com/
*https://shadow.tech/

Let's take a concrete example to understand the math. If a typical gamer plays 3D games mostly on Friday nights and Sunday afternoons, then the cost of buying a decent gaming PC ($1500-$3000) that sits idle most of the week may be too high. A game console ($500) is more affordable. But consider the $60+ per game price is where the console makers make most of their money. Streaming high-end games means this gamer can avoid that cost and pay only when they need it.

On the other hand, for the services, if 80% of their customers are on the same schedule as our example gamer, playing mostly on Friday nights and Sunday afternoons, then the service needs extra capacity to handle all of that concurrent demand, while that same capacity could sit idle 9-5 weekdays. That's very expensive for the service to maintain. Globalizing the servers for all time zones is one approach to evening out demand. But servers have the lowest latency near their customers, and worldwide latencies can be too high. Using the servers for other non-gaming applications (like AI training or molecular research) during the dips is the more likely way to make it profitable for the service over time.

**NVidia**'s cloud rendering service may have some built-in advantages in that regard, as their GPUs are undoubtedly cheaper to source and they already offer a variety of services, from off-line CGI rendering to AI training and simulation to take up the slack.

One thing holds true: cloud rendering is most likely centralized when the economies of scale tip the balance away from consumer hardware. Having everyone render their own virtual worlds locally is more decentralized, if not the lowest-friction approach. Even cell phones are quite adept at 3D graphics nowadays and limiting fidelity is one way to solve this — still, phones get hot, sometimes even just rendering video streams, and people don't want to kill their battery life. The one thing cloud rendering can never truly fix is the lowest possible latency for high-end XR, where 2-4 milliseconds matter.

A hybrid approach between cloud rendering and local rendering is often called "split rendering." **Mawari*** is one of the more visible examples in the wild (though there is evidence of related activity within companies like **Qualcomm**, **Ericsson**, and more). The key idea is to get both the lowest latency and lowest cost by cloud-streaming partially rendered 3D scenes. A mobile device does a small amount of final rendering to update the last-millisecond visual POVs for best head-tracking,  while the server does more of the heavy work and more shared computation, like for global illumination lighting and fluid simulation, which could be shared among multiple participants at the same virtual place and time. Commands could be pre-processed and streamed along with minimized data to smaller devices for final rendering.

Decentralized cloud compute is often called "edge compute," where the same kinds of network compute resources are deployed closer to the mobile endpoints for lower latency. For example, 5G cell towers might have compute and rendering resources embedded that can handle key tasks supporting mobile and AR devices. These machines would be shared among many users, and the handoff between mobile, edge, and cloud devices could become very dynamic.

In an even more truly distributed fashion, cloud rendering nodes could be provided by any network participant, from the metaverse user themselves (e.g., their home PC, laptop, or gaming console, connected to the network) to Metaverse Service Providers looking for micro-payments yielding a margin above the costs of operation.

To summarize, cloud rendering could help deliver really high-end user experiences on lower-powered devices, and it represents another possible entry point for MSPs to power the open metaverse Economy. But the cost structures need to become friendlier, we need new tech to break through the network overhead, and the creator/consumer value propositions need to be aligned to make the economics work. This is going to take time.

*https://www.mawari.io/

# 5.f Spatial Addressing and Mapping Schemes

Overall excitement around the metaverse, and peoples' energy envisioning products and services for it, inevitably lead to a series of big questions about how it is going to "take shape" –– in a somewhat literal sense. Questions such as:

- Is the metaverse one big shared, connected virtual space? Or several?
- How is / are space(s) subdivided and addressed? Are there multiple coordinate schemes? Is there an origin? If so, where is it?
- Who controls mapping services? Is there a centralized entity?
- Where does the map live? Is it even one map?

Thus far, no consensus has emerged around any of the above. Opinions abound and conversational volumes often escalate; this suggests that there will be a very long period of experimentation before the industry converges on solutions.

**Decentraland** and **The Sandbox** have made the first real strides in open Web3-based systems to define this — though "open" here means that there aren't too many barriers to entry — as long as a parcel is not claimed or it's for sale, all you need to claim it is some crypto to pay for it. But these schemes aren't open in the sense that there is not yet wide industry adoption for them. They are also tied to business models involving artificial scarcity, which is not necessarily appropriate to all metaverse scenarios. They are also heavily reliant on single providers of underlying infrastructure.

These models *do* show many characteristics of what could become future schemes for subdividing virtual space, offering ownership, and providing location services- when those capabilities make sense for the use case. The Metaverse Standards Forum and OMA3 have stated initiatives to standardize in these areas.

Perhaps subdividing blockchains via the use of subnets may hold the key to a future distributed, hierarchical addressing scheme in the spirit of DNS.

## Mapping and Navigating Virtual Space

An interesting aspect of the permanent record is found in the eventual interconnection of spaces. As the metaverse becomes populated with worlds, we should anticipate the need for ways to partition virtual spaces, connect them to each other, and support services to map and navigate them.

We could easily imagine the entire metaverse being in a single common coordinate system with (0,0,0) being the origin or center of it and the planet Earth existing virtually somewhere in that vast space. We often visualize our physical universe being organized as such, although in reality, any origin is equivalent to any other — there is no center, as far as we know.

In the virtual version, other virtual planets could exist, even whole galaxies, requiring some kind of virtual space travel to transit the metaverse. This is very unlikely to happen as simply as depicted above, simply due to the nature of decentralization — ad hoc placement over time results in a chaotic arrangement of elements.

Another issue to consider is "Which copy of a digital twin of the Earth exists at some fixed location?" There will likely be many digital versions or copies of physical worlds existing side-by-side (like the shards of an MMO) — but not in a way that we would tend to visualize as being literally adjacent.

The metaverse is more likely to come together like the web did, first by a series of independent spaces (or pages, for the web) and then interconnected in a much higher dimensional graph (or web) than our normal Cartesian 3D experience allows. It's probably safer overall to imagine a set of individual 3D spaces, each with their own origins, but with each likely sporting connections to other spaces.

We can call these connections portals, especially when they appear door-like. They need not be limited to rectangles or any 2D shape. When done right, they can stitch decentralized 3D spaces into an apparently consistent 3D space.

It's *consistent* in the sense that if two of us were standing on opposite sides of a portal, we'd have compatible views of everything that exists around us, nearby. But the mere existence of portals (like wormholes in space) can break our intuition about any paths through space. The shortest distance between points A and B is not necessarily a straight line as we're taught, but a function of any portals that form shortcuts or strange loops through multiple spaces on the way from A to B.

For example, you could be standing in a virtual room where the door to your left is a portal re-entering the same room on your right. You could travel in a straight line through these doorways forever and only ever cross the room again and again. What we mean by "consistent" is that anyone doing the same activity, with similar permissions to enter each door, would get the same experience as anyone else.

There is also a potential inconsistency over time, in that portals can change topology. A doorway you cross today may not be there tomorrow, which could add or remove a whole section of previously known or unknown terrain. As they say, "the map is not the territory," becomes even more literal now, as a single flat map may be impossible to draw.

The blockchains for LAMINA1 are where these many interconnections are stored over time, which is a major change from the web model, where hyperlinks are only stored in the documents, like web pages. The reason for this is that it requires dual permission to link two spaces –– each end of a portal must have permission to connect to the other end (web links are only one-way).

We wouldn't want someone transiting a portal into a space they don't have permission to enter. We also don't want to enable bad actors to make invisible portals that spy on us from remote locations. So the portal itself needs some basic permissions and disclosures for anyone nearby, as well as the "owner" of the underlying spaces. The other benefit of this change is that when spaces change, the configuration of portals can be better kept consistent because we can traverse them both ways and adjust.

As a related consequence of these design decisions, all spaces in the metaverse will likely have a unique name or URL, managed by a DNS-like system. Anyone can claim a new name for their space, as long as it's unique. There are potential problems with this type of namespace management, such as domain squatting; one possible solution would be to mandate that any such claim remain running persistently, with at least one functional supernode to answer queries. The timeout for names is the same as the time any supernode needs to retain its transaction history. So no new supernode can come in and impersonate the previous name-holder.

Within any of these spaces, the space owners are free to subdivide the space however they like, subject to the commonly agreed protocols.

# 5.g Tracking the Provenance and Life Cycle of Digital Assets

## Proof of Integration

While the progress in Web3 technology has been truly transformational over the past decade, blockchain and NFTs have thus far attempted to address “Who owns what?” but largely failed to capture “Who created what?” and “Who combined what and how much value was added?” Those systems will be essential to flow credit and revenues back to creators in a distributed marketplace.

While proof of integration (e.g. the ability to prove creative provenance easily & efficiently) and high-scale payments systems (e.g. the ability for metaverse builders to easily create rights payment groups of hundreds of thousands of recipients, safely pay them, and allow these recipients to cheaply receive their dues) will be a core component of the future Metaverse-as-a-Service stack on LAMINA1, they deserve their own whitepaper and a thorough framing of how existing systems try to tackle it.

This paper is currently under development.

**If you are developing protocols or infrastructure for rights, payments, messaging, or privacy, we would love to get your insights for the paper’s development. Reach out at ecosystem@lamina1.com.**

# 06
# CONCLUSION

"There's lots of people in it. You can interact with them in real time, no matter where they are. They're represented by audiovisual bodies called avatars, and they're having shared experiences that are fictional in nature. They're in fictional spaces, doing fictional things.

It's not all one unified walled garden. There are different bits of it that are created and maintained by different people."

– Neal Stephenson*

*https://www.ft.com/content/0ecab009-6543-4386-b936-0eecc9293d2e

In the introduction of this whitepaper, we set out to cover various capabilities and designs to support this vision that are being developed internally at LAMINA1, including:

- The LAMINA1 Blockchain
- General- and special-purpose LAMINA1 blockchain subnets (i.e. metachains) for flexible and scalable deployment of a variety of applications and experiences
- On-chain rights management and integrated with high-performance multiplayer simulation and world state storage services
- High availability, reliable asset storage and delivery
- Early experiments in blockchain-based identity management

Specialized metachains allow for the creation of blockchain-based applications and experiences in ways that overcome the limitations of today's monolithic chains. Additionally, they allow for a much wider variety of businesses and use cases than today's metaverse "walled gardens," empowering the users and the market –– rather than a centralized, risk-averse entity focused on specific business models –– to decide on the value of things that get made.

We also believe that the future metaverse also has to combine the best of immersive computing tech with the best of blockchain to ensure commercial success, longevity, transparency, and privacy online. On-chain rights management, distributed simulation, and storage services would help get us there, in a way that's accessible to all creators, not just a small club controlled by the few.

We also identified several system services that would benefit from an open, decentralized approach, and hope to engage partners to work with us to make those a reality, via an open approach with an eye toward eventual industry standardization.

All of this could one day lead to a world where metaverse experiences are actually interoperable, are accessible and usable, and are able to persistently scale, adapt, and grow – no matter the size or scale of the endeavor.

Which brings us full circle back to the mission of Metaverse-as-a-Service: To provide builders and creators with a usable, flexible, and decentralized framework to create the next generation of online worlds.

# 07
# ROADMAP*

## 2023

| | |
|---|---|
| **MaaS Whitepaper Release** | April 2023 |
| **Engine / Web Library Betas** | May 2023 |
| **Game State Alpha w/ Partner** | May 2023 |
| **Asset Storage VM Alpha (Subnet) LAMINA1** | June 2023 |
| **Betanet** | June 2023 |
| **Identity Litepaper** | June 2023 |
| **Asset Storage Beta** | August 2023 |
| **Messaging Alpha** | October 2023 |

*Roadmap is subject to change.

## Authors

Tony Parisi
LAMINA1

Will Carter
LAMINA1

Avi Bar-Zeev
RealityPrime



## Acknowledgements

Neal Stephenson
LAMINA1

Peter Vessenes
LAMINA1

Rebecca Barkin
LAMINA1

Casey Halter
LAMINA1

Alaina Bryan
LAMINA1

Alistair Hopkins
Hiro Capital


## Get Involved

Join the Community:
[discord.gg/LAMINA1](discord.gg/LAMINA1)

Visit the Website:
[LAMINA1.com](LAMINA1.com)


## Contact

**Ecosystem Development**
ecosystem@lamina1.com

**Events**
events@lamina1.com

**Opportunities**
jobs@lamina1.com

**Anything Else**
hello@lamina1.com


**Secret Code**
MaaSWh!tepaper

# DISCLAIMER

This whitepaper and the LAMINA1 website are intended for general informational purposes only and do not constitute a prospectus, an offer document, an offer of securities, a solicitation for investment, or any offer to sell any product, item or asset (whether digital or otherwise). The information herein may not be exhaustive and does not imply any element of a contractual relationship. There is no assurance as to the accuracy or completeness of such information and no representation, warranty or undertaking is or purported to be provided as to the accuracy or completeness of such information. Where the Whitepaper or the Website includes information that has been obtained from third party sources, the Translaminal team has not independently verified the accuracy or completion of such information. The whitepaper is a working document that is subject to changes as LAMINA1 is further developed and as it responds and evolves in light of new contributions and ideas, as well as potential changes in the regulatory environment that may govern Translaminal Inc or LAMINA1. You acknowledge that circumstances may change and that the Whitepaper or the Website may become outdated as a result; and Translaminal is not under any obligation to update or correct this document in connection therewith. The information set out in the whitepaper and the website is for community discussion only and is not legally binding. NOTHING HEREIN CONSTITUTES LEGAL, FINANCIAL, BUSINESS OR TAX ADVICE AND YOU SHOULD CONSULT YOUR OWN LEGAL, FINANCIAL, TAX OR OTHER PROFESSIONAL ADVISOR(S) BEFORE ENGAGING IN ANY ACTIVITY IN CONNECTION HEREWITH.

Nothing in the whitepaper or the website constitutes any offer to sell any L1 token (as defined herein) nor shall it or any part of it nor the fact of its presentation form the basis of, or be relied upon in connection with, any contract or investment decision. Nothing contained in the whitepaper or the website is or may be relied upon as a promise, representation or undertaking as to the future performance of LAMINA1.