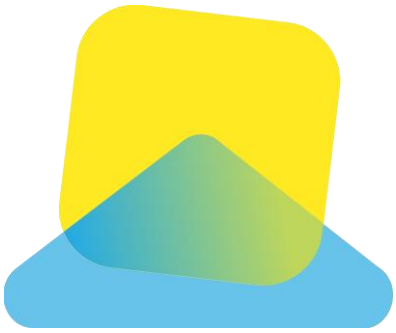


MUNDIS

the Multiverse Operating System

revision 3.4

flavius.burca@mundis.io

LEGAL DISCLAIMER

Nothing contained in this WhitePaper is an offer to sell or solicit any offer to buy any tokens. Mundis is releasing this paper only to receive comments and feedback from the public. If Mundis decides to sell any tokens, it will do so through legal offering documents, including a disclosure document and risk factors. Those documents are also expected to include an updated version of this White Paper, which may differ significantly from this version.

Nothing contained in paper should be treated or read as a guarantee or promise of the utility or value of the tokens or how the Mundis organization's business will develop. This paper outlines plans that could change at Mundis' discretion, and the success of which will depend on many factors outside Mundis control. These factors include market-based factors, factors within the data and cryptocurrency industries, and others. Any statements or assumptions about future events are based only on our (possibly incorrect) analysis of the issues described in this paper.

TL;DR

This is a one-pager high-level explainer of Mundis. It's an abstract aimed to get a quick understanding of the project before diving further into this Whitepaper.

This paper describes the building blocks required to achieve a scalable, interoperable Multiverse ecosystem (Metaverses, NFTs, GameFi) where users socialize, work, navigate, shop, and create experiences. The objective of Mundis is to deliver technology, standards, and protocols for current and future Metaverse projects:

- a Layer 0 backbone with sub-second finality times, inter-chain data exchange, distributed storage, identity management, naming system, and required supporting services for Metaverse projects
- Layer 1 byzantine state-machines with custom consensus rules, assets, and governance
- a Layer 2 extensible UI infrastructure where Metaverse projects deploy their visual assets, experiences and Metaverses

Developers use Mundis to build their Metaverse projects as Layer 1 byzantine state-machines, define their economic rules and interactions, issue their tokens, NFTs and assets. Metaverse projects that build on Mundis are natively interoperable and deeply integrated in the ecosystem. Tokens and NFTs can be used across the entire Multiverse, teleported to/from other blockchains, and integrated natively in the UI.

Mundis is both a technology and visual platform, delivering a complete operating system for Metaverse projects to thrive. It provides all required tools and services for technical and economic exchange, payments, DeFi, visual asset and game development. It's a complete ecosystem where validators participate in Layer 0 and Layer 1 consensus to validate transactions and secure the Multiverse.

A. Introduction

1 A state of things

The blockchain industry grows exponentially due to the large influx of new dApps, tokens, games, and NFTs published daily. Businesses adapt their products and services to get a small share of the crypto space. The number of actors and data points moved through blockchain systems will soon reach significant figures. Existing blockchains must support this demand and still provide high throughput, low latency, and intelligent data archiving.

A system that provides many transactions per second (TPS) creates a great user experience and increases business applicability. In addition, most consumers demand near-real-time transaction processing; they cannot wait minutes for a transaction to be approved.

A high number of TPS is an absolute requirement for the increasing volume of information and interactions. Current blockchain systems have a sequential nature, which limits scalability. Great projects like Solana, Elrond, and Algorand lead the TPS battle with innovative sharding and consensus optimizations. Besides sequential processing, existing systems have one thing in common: they are one-size-fits-all systems bounded by the chain's consensus rules and economic system.

The Multiverse has some specific requirements, given its multi-world nature and complexity of interactions between its worlds and users. Until now, the concept of parallel task processing has been adopted in a minimal to increase efficiency. Sharding is the first attempt at parallel processing. However, it is only a primitive approach with many challenges like inter-shard transaction processing, fast consensus between validators, cross-shard smart-contracts, and more.

To achieve a parallel, multi-world distributed architecture, a different approach is needed, and that approach would be based on a parallel, interoperable, distributed multi-chain architecture. Notable projects like Cosmos and Polkadot are leading the multi-chain battle. Cosmos uses the Tendermint consensus, which is a protocol that provides instant finality, producing and finalizing one block at a time. However, because it's a classic pBFT algorithm, it has $O(n^2)$ transport complexity and can only finalize one block at a time [1]. Polkadot's Relay Chain will have a maximum of 100 slots for Parachains. Out of 100, a few are reserved for Parathreads and other dedicated Bridges (Bitcoin), leaving even less for users. This is obviously not enough. The Metaverse is ultimately an ecosystem of infinite systems that must exchange information reliably and perform interactions in a non-repudiable way.

2 The Metaverse

Think of it as a virtual world. It can be a computer-generated world or an augmented reality environment. It is a place anyone can visit and interact with using a computer, smartphone, or specialized VR/AR device. However, most importantly, there won't be a single Metaverse. Instead, as with any new concept, many companies will push their version of the Metaverse with specific benefits, features, economics, graphics, collectibles, and assets. As a result, we will have many Metaverses to choose from: gaming worlds, infinite maps, workplaces, social environments, and much more.

Ideally, a Metaverse will depend on a blockchain architecture. This is only natural since a blockchain provides the means to implement monetary value in the Metaverse. A suitable blockchain would need to be flexible and fast enough to implement and execute the rules of the entire Metaverse. In addition, it would need to interact with other blockchains, so users of the Metaverse could benefit from all things already built on those chains: NFTs, games, tokens, smart-contracts, data oracles, scaling solutions, and more.

It would be an ecosystem with the proper scalability, performance, and instruments to accommodate the Metaverse. An ecosystem with its own rules, economy, assets, and interactions. An ecosystem you can navigate, explore, and visualize. An ecosystem you can be part of.

We postulate several requirements a new ecosystem must have to accommodate a Metaverse:

- it needs to have a single timeline that serves as a reference system for all the events in the Metaverse
- it must accommodate an infinite number of worlds with their own genesis and consensus rules, economies, and assets
- all worlds in the Metaverse must be able to interact to exchange assets and monetary value
- worlds will have custom rules, economics, graphics, and assets. World developers need the appropriate means and tools to create everything they can imagine
- it must be able to exchange assets and information with existing blockchains: NFTs, dApps, and tokens
- it must be able to exchange assets and tokens with other Metaverses
- it must be able to interact with the external world (e.g., data oracles)
- members of the Metaverse must have an identity and means to advertise their resources (NFTs, tokens, worlds)
- it must be navigable and provide visual means to explore the Metaverse
- it must be highly secure

3 Market overview

There are various types of metaverse systems already available. We can broadly classify them according to their governance and purpose.

Depending on their governance structure, we observe there are centrally owned metaverses like Fortnite or Facebook's future Meta, and metaverses with decentralized ownership, like Decentraland, Axie Infinity or The Sandbox.

According to their purpose, we can find metaverses strictly for gaming, like Fortnite, or game-hybrids, like My Neighbour Alice, that add a tokenization twist with play-to-earn mechanics, or The Sandbox - a tokenized game builder-platform. All examples allow in-game socialization and marketing integration. Another representative category are metaverses focused solely on opportunity investments, like Polka City - where you just buy and rent land. There are also hybrids like Decentraland or Bloktopia, that work both as opportunity investment, social, and game experience platforms.

The most recent hype on the matter was ignited by yet another metaverse category: the collaborative work metaverse, suggested by Meta's new product announcements. This attention shift brought up other contesters like Mesh for Microsoft Teams, or Salin. All these are VR engagement platforms, designed to be experienced with a VR headset

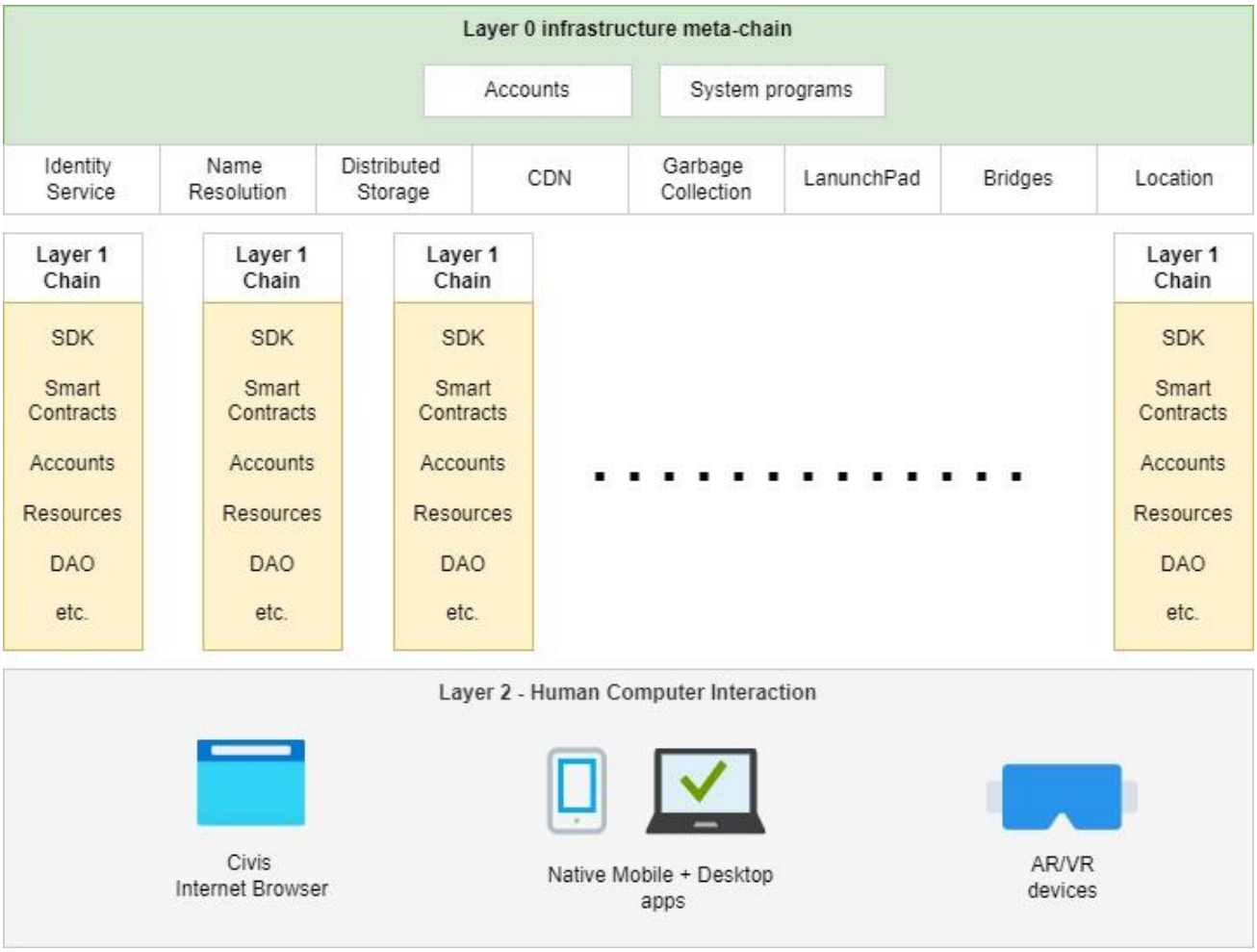
Departing from the classic model, Mundis is a decentralized platform that uses tokenization and core blockchain technology to drive product engagement, gather attention and enable governance.

4 Mundis

We believe the Metaverse's vast benefits are untapped yet. We believe it can become a mainstream medium of engagement, servicing all user categories through useful tools and thrilling experiences. And we believe this will be achieved well before VR/AR technology becomes affordable and advanced enough for the mainstream market.

Using standard 3D desktop and mobile environments, we propose a multiverse ecosystem of parallel, interconnected worlds, designed for massive scale, extreme performance, and unlimited extensibility. An ecosystem where builders and citizens are equipped with all necessary tools and means to build on and immerse in a variety of Metaverses, using their desktop or mobile phones.

This ecosystem is hereafter named Mundis. Since we are building a multiverse, Mundis is not only a blockchain, but a layered architecture of interoperable worlds, services, and human interaction devices.



Each layer provides specific services to a Metaverse, from critical infrastructure to end-user interaction. The layers presented below are what we identified to be the starting point of our

Multiverse. As the system evolves, new layers might be added to accommodate future requirements:

- Layer 0 – An infrastructure chain that provides a global, forward-moving clock, checkpoints L1 chains, handles positioning and global navigation, and drives governance for the Multiverse. It has a fast consensus mechanism designed for speed and sub-second finality, and native support for Multiverse services: Identity Management, Name Service, Decentralized Storage, CDN, Launchpad, Bridges, Garbage Collection, Location, etc.
- Layer 1 - Interoperable worlds, created as specialized, byzantine state machines with PoS-based consensus. Each chain has its own genesis and consensus rules, optional smart-contract support, DAOs, and more.
- Layer 2 – the Human-Computer Interface (HCI) to interact with the Metaverse using desktop, mobile and AR/VR devices. It's an engaging visual, interactive experience for users to immerse themselves in the Multiverse.

B. The Backbone

The Backbone is the blockchain infrastructure of Mundis. It governs all transactions that happen in the Multiverse and all its Metaverses, handles consensus between L0 and L1 validators and keeps a verified history of all L1 worlds. The Backbone is a development platform that will evolve to be an ecosystem of interconnected worlds.

1 Layer 0

The role of L0 is to coordinate the Multiverse and all associated Metaverses. It's a blockchain based on a global timeline of events with an optimized pBFT replicated state machine that can do sub-second finality times. It runs on a globally distributed "clock" with Proof-of-Stake (PoS) consensus, built to address the needs of our Multiverse.

The global "clock" is not a consensus algorithm but a computational algorithm that provides a way to cryptographically verify the ordering of events to solve the agreement on global time. It's a VDF hash-chain used to checkpoint L1 worlds and coordinate global consensus. It allows near-instant finality up to hundreds of thousands of transactions per second.

The L0 chain needs actors that can process transactions and participate in consensus. These actors are named validators in existing blockchains, but in Mundis, we call them L0 validators.

L0 validators are the keepers of the Multiverse. Besides validation tasks for transactions and consensus, they also participate in global governance. Therefore, it is desirable to have many L0 validators for the ecosystem to be decentralized, and Mundis does not limit this number as some existing blockchains do. As the Multiverse grows, more validators will be needed to increase the performance of the ecosystem.

1.1 Delegated Proof-of-Stake

Layer 0 is a Proof-of-Stake (PoS) based blockchain forked from the Solana codebase. We chose Solana as a starting point because of its fast block times, Proof of History innovation and

very high throughput. Building on Solana itself was not possible because it would require changing the Solana main-net to accommodate all changes required for Mundis-specific features.

To become an L0 validator, a participant needs to stake tokens to become eligible. A PoS network is one in which a set of trustless entities can achieve consensus (i.e., agree) on the validity of a sequence of transactions. Research on this topic demonstrated that consensus could be reached if less than 1/3 of the validators are dishonest (Byzantine).

The network may be stalled if dishonest validators exceed 1/3 of the total number of validators. Dishonest validators may simply decline to participate, preventing the remaining validators from achieving the required 2/3 super-majority. In this case, the network can no longer produce transactions. If dishonest validators account for 2/3 or more of the total nodes, they can conspire to make any form of arbitrary transaction. This is the worst-case situation because the network can no longer produce actual transactions but only those injected by the dishonest 2/3 super-majority.

Ideally, Mundis will have many validators to promote decentralization and to make corruption economically unfeasible.

Proof-of-Stake introduces the idea of "stake" to the design of consensus-based networks, which means participants have "skin in the game." The "stake" can be forfeited if they are found to be misbehaving. This is called slashing and enables the network to be designed more efficiently with penalties for validators that engage in bad behavior. It's important to note that slashing isn't a substitute for security. Slashing merely allows for repercussions for improper behavior by validators that would cause the network to be momentarily disrupted or inefficient.

Proof of Stake in Mundis confers varying degrees of authority on the validators who secure the Metaverse. Each validator has a different voting power based on the value of their active investment. As a result, if validator A votes for consensus with twice as much stake as validators B and C, this is comparable to both B and C voting for consensus.

With slashing, validators are discouraged from bad behavior that would negatively impact the network. Slashing also reduces a validator's influence on the network by reducing his stake-weight. It can also encourage users to find a more honest validator to stake their funds in. For example, if a validator choses to submit more than one block at a time, the network will consider it bad behavior. It will create two forks, and the network must choose the right fork to commit to the ledger. Even if the other validators eventually commit the correct block, the additional cost of fork resolution will cause a slow-down. This is considered bad behavior, and the validator that submitted the extra block will be slashed.

Mundis citizens can choose trustworthy validators that provide the best service to the network. If a validator is honest, was never slashed, and has a good relationship with its stakeholders, it should receive more stake than an validator who lacks one or more of these characteristics. The staking mechanism is designed to favor well-behaved validators, thereby boosting the ecosystem's performance and safety.

L0 validators are rewarded according to their stake. This means it's in their best interest to attract and retain stake by showing good performance and returns, transparency, honesty, and a good relationship with his stakers. This means that validators' financial compensation is aligned with the network's goals.

An validator must not receive a large portion of stake, because that would put too much authority in one node and create centralization. Also, a very powerful validator might be hacked or could unexpectedly change its behavior in a bad way ("go rogue"). Mundis imposes high-order limits on how much stake a validator can attract. This largely depends on the

number of active validators and is a mechanism designed to prevent centralization of wealth and emergence of super-validators.

1.2 Global Timeline

Mundis has a single, global timeline for the entire ecosystem that gives a verifiable, exact order of events that happen in the Metaverse. It's a universally synchronized "clock" all validators use to synchronize themselves with the rest of the network. This clock is not based on actual time (i.e., seconds) because that would require agreement on the actual time and trustless synchronization of time-based clocks throughout the Mundis network.

Methods to synchronize time between remote computers have been out there a long time, but those methods require communication and agreement between nodes. Also, a dishonest node can ignore his turn and claim his clock is the correct one and everyone else's clock is wrong.

Time sync issues can be resolved through a variety of techniques that establish trustless synchronization of time-based clocks in a network. The problem is that their accuracy also degrades their performance a lot. Imagine a solution that simply creates slots that are one year long. If a validator receives a block that has a wrong year, it may assume the leader that proposed the block has a problem with his clock since it's unlikely the validator has a clock that is one year off. Also, if the interval is short, a validator could have no way to determine whose clock is skewed, as network agreement would take longer than the actual clock skew. Without an agreement, validators cannot properly order the block proposals in a way that that they don't overlap or collide.

To circumvent clock synchronization issues, Mundis does not use real clock time or "wall clock time." Instead, it uses a "clock" based on the time it takes to run an SHA-265 VDF (Verifiable Delay Function) a fixed number of times. This was originally proposed by Solana as Proof of History. All computers using the fastest generally available processors will be limited to computing a fixed number of SHA-256 hashes per second [6]. This looping VDF that runs a fixed number of times can be said to closely approximate a second's duration.
Mundis measures time in "ticks," which are the number of times the VDF loops, approximating a fraction of a second. All validators run this "clock" on their node by running the looping VDF as quickly as they can. The actual Metaverse time is represented by values produced by the VDF. By providing an initial value I and a value computed after N iterations starting from that initial value, a validator can measure how much Metaverse time passed between N and I by running the looping VDF.

It is important to remember that Multiverse time is measured in ticks and was not designed to approximate human time, but to be the global reference "clock" for the Multiverse. L1 chains have no control over global passage of time.

1.3 Consensus

Layer 0 must be fast, both in terms of overall throughput and latency. While Proof-of-Stake can guarantee consensus if the super-majority of nodes are honest, it cannot guarantee that it occurs quickly or with high throughput. Mundis builds on Solana's Tower BFT asynchronous implementation that relies on the Multiverse timeline for fast consensus.

The algorithm offers both liveness and safety if at most $\frac{n-1}{3}$ nodes are byzantine. Mundis is an asynchronous distributed system where nodes are connected by a public network. The network may delay or fail to deliver messages, duplicate them, or deliver them in a different order. If replicas are unable to execute a request, they must move to a new view to provide liveness.

The time period when the super-majority 2n+1 of non-faulty replicas are in the same view must be maximized, and this period of time must increase exponentially until the requested operation executes [7].

Blocks are chained together to create a cryptographically provable sequence. By receiving 66% of stake-weighted votes, any block can "prove" it's part of L0. The sequence of submitted blocks should be ordered; otherwise, multiple nodes will submit blocks simultaneously, and computing the order of blocks will be a costly and time-consuming process based on fork resolution algorithms.

This is where the concept of "leader timetable" comes in. The set of validators that could vote is known before a round starts. Validators "register" as voting nodes, and stakers assign (delegate) their stake to voting validators, allowing those validators to cast stake-weighted votes to select valid blocks. The set of validators that can submit blocks is known beforehand, so it's possible to organize the network in a way that only one validator is allowed to submit one block at one time.

Each block is produced during a slot, and each slot in the leader timetable is allocated to one validator. Thus, the leader timetable provides an ordered succession of validators, and only the validator that has the allocated slot in the timetable can propose a block during that given slot. This way, the validator already sorted prospective blocks, making it easy to detect and reject an out-of-order block proposal.

For example, consider are four validators (A1, A2, A3, A4) and the following leader timetable:

$$A1 \rightarrow A4 \rightarrow A3 \rightarrow A2 \rightarrow A4 \rightarrow A1 \rightarrow A2 \rightarrow A3$$

A1, A2, A3, A4 must submit blocks exactly in the order above, with the sole exception a slot might be skipped if the validator assigned to that slot is unavailable or misbehaving.
Slots are organized into epochs. The leader timetable for the current epoch E_n is determined by data from epoch E_{n-2} . Therefore, the leader timetable for epoch E_n is known for the duration of one full epoch, which is plenty of time for everybody to know the leader timetable.

The leader timetable must not be the same for the epoch E_{n+1} and it must be impossible to determine in a deterministic way. Mundis uses a random number generator to select validators based on their stake-weight, performance, and availability. While validators will get slots roughly in proportion to their stake-weight, the exact slots a validator will have are non-deterministic and impossible to predict.

A random number generator is seeded with the previous state of the network to compute the leader timetable at the end of epoch E_n. The generated random values are not predictable or controllable by any one validator because they come from the agreed state of the network, which cannot be intentionally coordinated or enforced. Because the leader timetable is always computable from data that is available to every validator without requiring any network communication, each validator can independently compute the leader timetable for epoch E_{n+1}.

This is a significant performance enhancement because it eliminates a round of network communication between validators.

Once a leader timetable has been set, it is up to each validator to adhere to it and propose blocks when "their turn" comes around, as specified by the timetable. But what prevents validators from emitting blocks in the wrong order? L0 validators rely on the Multiverse timeline to synchronize on half-second boundaries. They query the VDF "clock" once per tick, which provides a sequence of numbers for each slot representing the total time spent in a slot. This enables the sequence to be invalidated more rapidly if it turns out to be incorrect. For instance, sampling ten values per slot enables a validator to accept or reject sequences faster, as each number may be examined in parallel in 1/10th the time it took the original validator to compute the sequence. This is an inherent property of VDFs, and it means legitimate validators cannot be "flooded" by dishonest validators. If validation would take the same time as the production of values, a bad actor can submit many bad blocks and cause other validators to spend a lot of time verifying these blocks. If blocks can be checked much faster than it takes to produce them, an attacker needs to spend a vast number of resources to generate this kind of attack, making it impractical.

The leader streams blocks as it verifies transactions. Transaction validation and streaming to neighboring validators are done in parallel to decrease latency. The timeline's VDF begins with the last SHA-256 value computed from the previous block's timeline. In the new block, the validator includes transactions, and the timeline sequence values. When his allocated slot ends, the validator completes the block, signs it, and streams it to the network for validation. Other validators can easily verify the proposed block's timeline value and slot using the data they have before they vote on the block. The next selected leader repeats the entire process by doing the same thing using the previously emitted block as its starting point. The block also keeps the signature of the leader who proposed it, allowing other validators to prove that block for that slot was proposed by the correct leader. This allows validators to follow in leader sequence with minimal overhead, contributing to the speed of the network. The leader timetable is censorship-resistant, which means a validator cannot exclude another validator by emitting a competing block in another leader's slot.

Since Mundis has an embedded global clock, the pBFT timeouts can be encoded in the ledger itself using a two-step process [8]:

1. when casting a vote, it has an initial timeout of N timeline hashes. A validator guarantees with slashing that once a vote has been cast, he must not vote only for timeline child hashes of that vote, for at least N hashes
2. the timeouts for all the previous votes double. Voting is limited to a predetermined number of slots. Each slot has a potential rollback point, but each subsequent vote doubles the length of time required to stall the network before it can unroll that vote. Adding new blocks to the ledger increases the likelihood of previous blocks being confirmed. The number of slots to which old votes are pledged doubles with each slot. Once a super-majority votes on a timeline hash, that hash becomes canonical and cannot be reverted.

To avoid being isolated from the rest of the network, each validator guarantees that they vote only when a super-majority of the network votes on the same ledger. Each validator keeps track of the timeout period an ancestor vote exceeds a predetermined threshold and ensures a super-majority of the network has voted on a fork containing that vote.

In practice, validators check if the super-majority has voted on a slot that will commit to a timeout of 10 minutes, and if not, they don't vote.

When timeouts start to happen, expired votes are cleared, and timeouts double for ancestors if the child has an equal timeout.

This technique enables the network to stream blocks continuously without blocking the ledger until a super-majority observes the same ledger. Every validator can compute the timeouts for every other validator from the timeline itself without any network communication, making the BFT implementation asynchronous in nature.

Micro forks are quickly discarded. When a validator detects multiple forks, honest validators compute the effective stake-weighted timeout of every fork and pick the heaviest one. The network generates rewards only for votes that reach the 2^{32} timeout. Therefore, validators are incentivized to vote for the heaviest fork since the fork with the largest amount of stake-weighted timeouts will generate the largest amount of rewards.

Let's assume there we have leaders A1, A2, A3, and the leader timetable is $A1 \rightarrow A2 \rightarrow A3$. Let's see what happens if A3 is malicious and decides to propose a block at the same time as A2. A3 will have to create a new fork based on A1's block, but the leader schedule says that A2 is the next leader after A1. If A3 emits a block that has the same timeline value as A2, other validators will know A3 is cheating because it proposed a block during A2's slot. To make it more believable, A3 must forge the fact that A2 never proposed a block. If A2's block is missing, A3 must first spend A2's slot to produce a timeline proving it waited for the full duration of A2's slot before beginning its own slot. Only then would it start emitting its own block, including the next timeline sequence. Other validators will accept A3's block as valid since A3 is the new correct leader in the correct slot. Therefore, A3 must compute enough timeline hashes to show it waited for A2's slot before proposing its block. In the meantime, A2 already submits its block that forks off A1's block. From the perspective of the network, the following events occur:

1. A1 completes a valid block with the correct timeline value.
2. A2 immediately starts streaming its new block, and the network sees that A2 is active and producing a block properly forked of A1 and with the correct timeline value.
    i. if A2 is faster at computing timeline than A3, A2 will complete its block the same time A3 starts emitting its block, which is a fork of A1's block. By this time, the network saw A2's block, so it rejects A3's block. Due to network latencies, some validators may get both blocks and may not be able to tell which one to vote. In this case, the super-majority would have to see A3's complete block well before any of A2's block to accept it.
    ii. if A2 is slower at computing timeline than A3, then A3 can send its block before A2 finishes. It is unlikely for A3 to be much faster than A2 so it can begin its block before a significant fraction of A2's block is already done and seen by the network

The relative speed of A3 against A2 decides how A3 can censor A2. Still, A3 would have to be much faster than A2 and have any chance of censoring A2. This is because all validators can compute the recursive VDF in comparable time, so it's unlikely for A3 to be so much faster than A2 to compete with him for a block in his slot.

If A3 loses, it will not be allowed to propose a block for its slot. This is because once A3 proposes a block that it's a fork of A1 (pretending A2 never produced a block and was skipped),

if it loses (and chances are he will), any attempt to propose a different block forked off A2 for his slot, means producing two blocks for A3's slot (an A1 - A3 block and an A2 - A3 block). This is considered misbehavior and will result in slashing for A3.

Therefore, A3's attempt to censor A2 by competing in A2's slot will fail with a high probability. For every failed attempt, A3 loses out on any chance to emit a block during its slot. On balance, it's a losing strategy for A3 most of the time, so he is very unlikely to try it. In summary, L0 is a Proof of Stake network that works reliably if more than 1/3 of nodes are honest. In addition, the network's speed is increased through the natural fork avoidance mechanisms built into the leader timetable and the global timeline mechanism that ensures that the leader timetable is enforced while maintaining high speed and low latency.

1.4 Layer 1 sharding

Mundis is a dynamic ecosystem where all L1 chains are grouped into shards and asynchronously attached to L0. The proposed approach is designed with the following objectives:

- High scalability: increasing or decreasing the number of L1 chains should not affect the performance of L0
- Traceability: L0 should determine in a deterministic way the grouping of L1 chains into shards
- Efficiency: shards should be balanced appropriately between L0 validators, and an L1 chain can be Garbage collected once it loses the support of its validators

L0 constantly receives hashes of finalized blocks from L1 chains. Finalized block hashes are stored in the L0 global timeline to keep up with the state of all L1 chains and vice-versa. Communication between L1 and L0 chains is made using message queues and an extension of the XCM format. L1 chains need to send at least one valid block per L0 epoch. Failure to do so will cause the L1 chain to be "jailed" and his L0 genesis stake to be slashed by 0.01%. A jailed L1 chain will not receive any inter-chain transactions. This process will continue until L0 receives a valid block from the L1 jailed chain.

At the end of an L0 epoch, L1 shard reorganization and garbage collection are triggered to rebalance L1 chains. Rebalancing will garbage collect (prune) obsolete L1 chains and possibly re-group them into new shards.

A special note: Mundis assumes that L1 chains are byzantine by default. L0 does not verify or execute any L1 transactions, so it is the responsibility of users to ensure an L1 chain is trusted before exchanging assets with that chain.

1.5 Global services

Some critical services are needed for a Metaverse to work. These services are hereafter named L0 services. L0 services are deployed by L0 and run inside validator nodes. They are native L0 system programs that can be consumed by L1 chains or external actors. Consuming an L0 service might incur fees, depending on the nature of the action. Actions that alter the state of the ledger will have transaction fees because they require transaction validation.

1.5.1 Identities

Identities in the Multiverse are not real identities, although they can be if a participant chooses to do so. An identity is simply a means for the participant to create a public profile so he can be identified in the ecosystem; a profile must have a name and optionally an avatar. Mundis has several roles for participants in the ecosystem:

- L0 validator – validates L0 blocks and participates in global governance
- L1 validator – validates blocks for an L1 chain and participates in the governance of the L1
- Citizen – has an active role in the Multiverse or in a specific world

L0 and L1 validators are required to have a Multiverse identity because they perform block validation. It is in their direct economic interest to be trusted by citizens, and the more trust they have, the more stake-weight votes they get, increasing their chances of receiving block rewards.

Participants become Citizens by getting an identity in the Multiverse. Some L0 services might be restricted only to Citizens (e.g., LaunchPad), but an L1 chain can also make its services available only to Citizens if the builder of the L1 chooses to do so.

The Identity Service is not a full-featured identity management solution and is not designed to be so. At its minimum, it keeps a global mapping between L0 addresses and user profiles.

1.5.2 Name Resolution

Like the Identity Service, name resolution provides friendly names for machine-readable identifiers such as addresses, content hashes, and metadata, like current Internet domains. The Name Service supports forward and reverse-address resolution, making it possible to associate metadata such as canonical names with Mundis addresses. The Name Service is like DNS, the Internet's Domain Name Service, but has significantly different architecture due to the specifics of blockchains and the Multiverse. Like DNS, the Name Service operates on a system of dot-separated hierarchical names. The owner of a domain has complete control over sub-domains. It provides similar functionality to ENS (Ethereum Name Service).

The name resolution service uses the top-level domain (TLD) (e.g., ".meta "). Worlds and assets (tokens, NFTs) can get friendly names so they can be easily referenced, advertised, and found by other participants. In addition, name resolution plays an important part in Layer 2 - Human-Computer Interface (HCI), where the Mundis internet browser will use the Name Service to resolve Multiverse addresses based on their friendly names, creating an Internet-like user experience for navigating the Multiverse.

Like ENS, the Name Service has two components:

- Name Registry – maintains a list of all domains (and subdomains) and stores some critical pieces of information about each domain: the owner of the domain, the resolver, and the TTL for all domain-specific records
- Resolvers – translate names into L0 and L1 addresses

Domain owners can transfer the domain to a new owner or change ownership of subdomains. L0 will have a system resolver for all L0 addresses, but L1 chains need to implement their own resolver to translate names into L1 addresses.

The Mundis Name Service is described in reference specification MDS-137.

1.5.3 Positioning System

Mundis has the notion of multi-space which are variable size 2D planes that have their origin at (0, 0) and expand to (W_i, H_i), where W is the width of a plane and H is the height. A location is basically a unique set of coordinates (x_i, y_i) that identifies a spot in one plane, where:

$$\begin{cases} 0 \leq x_i \leq W_i \\ 0 \leq y_i \leq H_i \end{cases}$$

Based on simple 2D geometry, we can compute distance, area, perimeter, and other location-specific indicators in a plane and give routing instructions to a specific place.

The number and size of initially available 2D planes will be initially set in L0 genesis and can increase or decrease in the future if L0 validators vote to do so.

Each L0/L1 asset can have a location and size to be positioned and identified in the Multiverse.

1.5.4 Storage and CDN

IPFS provides a high throughput content-addressed block storage model with content-addressed hyperlinks. IPFS is peer-to-peer; no nodes are privileged. IPFS nodes store IPFS objects in local storage. Nodes connect to each other and transfer objects. These objects represent files and other data structures [5]. L0 validators are IPFS nodes identified by the cryptographic hash of their Mundis address.

Developers decide how much data will be stored in Distributed Storage, how long, and at what cost. Once a data object is committed to the Distributed Store, L0 validators will pin the objects in IPFS for permanent storage and receive rent for doing so.

To deliver fast Distributed Storage resources, L0 validators will also host CDN services to accelerate IPFS file reads. This creates a geographically distributed content network that serves content to users by bringing it closer to where they are. CDN nodes use caching, a process that temporarily stores copies of files, so that you can access Multiverse content from an Layer 2-enabled device faster, using CDN nodes that are close to you. CDNs can cache any type of content like web pages, images, videos, audio, and binary files. As a result, users can watch a movie, play games, and maybe check their NFTs without having to wait a long time for content to load.

Objects in the Multiverse can have persistent data resources as binary large objects (BLOBs). Distributed Storage users need to incentivize L0 validators to pin BLOBs objects on IPFS.

BLOBs open a new world of possibilities for builders to create distributed, decentralized applications like games, databases, data-sharing networks, and much more. L1 creators can host their binary data in public cloud services like AWS or in Mundis. The main difference is that storing BLOBs directly in Mundis enables true decentralization and trust. BLOBs published in the Anima Distributed Storage are public, immutable, and hard to censor by central authorities like AWS, Azure, GCP, etc.

BLOBs play an important role in Layer 2 because they can be (but are not limited to):

- static web content (HTML, CSS, and JavaScript). Developers can create simple websites, dApps, WebGL games and GUIs. Layer 2 will render Mundis web content.
- 2D/3D assets developed using the toolkits of choice (e.g., Unity, Unreal Engine) or other libraries

1.5.5 Garbage Collection

Garbage collection is a form of automatic resource management widely used in software development. The garbage collector attempts to reclaim memory occupied by objects that are no longer in use. Mundis does garbage collection (or pruning) to prune L1 chains that are no longer supported by L1 validators. Garbage collection is executed at the end of each L0 epoch and can trigger L1 shard reorganization.

1.5.6 Bridges

A cornerstone technology of blockchain interoperability is the blockchain bridge. Blockchain bridges are ways for two economically sovereign and technologically diverse chains to communicate with each other. Bridge designs come in various flavors ranging from centralized and trusted to more decentralized and trustless [3]. Decentralized, trustless bridges are implemented using system programs for bridging with Turing-complete enabled blockchains like Ethereum and Solana or specific protocols like XCLAIM [4].

Mundis goes one step forward and provides system-level services for bridging with current notable blockchains like Ethereum, Binance Smart Chain, Solana, Elrond using Simple Payment Verification (SPV).

SPV is a term for a set of methods used by light blockchain clients to verify network state without fully storing the entire chain. It's a kind of Merkle-proof used to verify the presence of a transaction in a block by comparing it against the root hash in that block's header. This allows reaching a probabilistic level of certainty about on-chain events with minimum trust required in respect to network nodes. SPV services are essentially system programs deployed on L0 for each blockchain.

Wherever possible, SPV services will use existing blockchain layer 2 solutions to accelerate transaction processing. One very good example is the Polygon protocol for Ethereum.

2 Layer 1

A L1 chain (or side-chain) is a business-specific application that runs in the Multiverse (i.e., a world or a Metaverse). They are smaller, independent chains that manage their own consensus and have control over how blocks are generated. L0 does not verify or execute L1 transactions, and it only notarizes valid L1 blocks through the timeline. L1 chains are independent, interoperable application systems that benefit from Anima Services. Inter-L1 communication is achieved using messages routed through L0.

The main feature of L1 chains is the ability to move assets between them, enabling complete interoperability of Metaverse worlds and asset interchangeability. Assets are designed to be fungible like coins or tokens or non-fungible such as NFTs.

Consensus inside L1 chains is Proof of Stake (PoS) with an optimized BFT state-machine designed for high speed and fast finality, enabling fast block production and finality. This is important because L1 chains need to support a wide range of business and consumer applications, from DeFI systems to games.

L1 chains can be developed using Substrate as the main SDK. Substrate takes a modular approach to chain development by defining a rich set of blockchain primitives for developers to use. L1 developers can create their own custom runtime using the Framework for Runtime Aggregation of Modularized Entities (FRAME) and pallets available in Substrate or directly with Substrate Core.

Upon genesis, an L1 must be registered in the L0 registry, a database-like construct that holds both static and dynamic information on each L1. Part of the registration process is to generate an identity keypair for the new L1, create an L1 genesis account in L0 and deposit an amount of L0 tokens in the respective account. The identity keypair will be used in the genesis configuration of the respective L1 chain.

L1 operations include the suspension and removal of L1 chains. Suspension can happen only if the L1 chain account has no more Mundis tokens available in their L0 account registry, and the owner of the L1 chain must add funds to reinstate his chain.

The removal (garbage collection) of an L1 altogether would come in the following circumstances:

- The L1 is suspended for a long period of time.
- The L1 doesn't have validator support anymore and is not producing blocks
- Through an L0 referendum: this should be an emergency measure only, available for rogue L1 chains that demonstrate malicious behavior like spamming other L1 chains with inter-L1 transactions.

2.1 L1 validators

L1 validators are nodes that maintain an L1 chain by validating transactions and participating in the consensus of that L1. Since an L1 has Proof-of-Stake (PoS) consensus, validators must stake Mundis tokens to become eligible validators for an L1 chain.

It's up to the creator of an L1 to recruit validators for its project, and Mundis provides the LaunchPad to help. Creators can list new projects for free on the LaunchPad, and citizens decide whether they pledge support for the project or not. A project that has citizen support has greater chances to attract validators.

The economics of an L1 is also a decisive factor to consider for validators. The creator must carefully design the economics of its new L1 to be attractive (i.e., validators must be rewarded for validating blocks).

2.2 Ruling system

When creating a new L1 chain, builders can set genesis rules that enforce how the L1 will work. Genesis rules define bootstrap parameters and consensus rules. Bootstrap parameters set the initial state of the L1, the configuration of the economic system (e.g., tokens issued, inflation, etc.), and access to L0 Services, DAO enablement, etc.

Consensus rules, on the other hand, configure how block finality is achieved and what happens when a block is finalized. Rules are flexible to allow any use-case to be implemented. Consensus rules also determine the minimum number of validators needed to support the L1 chain. If the L1 chain does not gather or maintain enough validators support, it will be Garbage Collected by L0 to free up Metaverse resources.

The builder of an L1 chain can choose a DAO model for governance. The validators will be included in the DAO and have voting rights on the L1 consensus rules. The DAO model promotes active participation in the life of a World and creates decentralization. validators will probably prefer to join an L1 chain with a DAO model as extra insurance for their stake.

2.3 Assets

An L1 chain can have both fungible and non-fungible assets. Assets are designed to be fungible like tokens or non-fungible like NFTs.

The main goal of Mundis is to promote interoperability. Exchange of assets must be supported between L1 chains and between an L1 chains and L0. We chose Parity's cross-consensus message format XCM to enable native exchange of assets with XCM-enabled blockchains.

XCM provides different methods to transfer assets between chains, as stated in the Cross-Consensus Message (XCM) Format specification [10]:

- *Remote Transfers: control an account on a remote chain, allowing the local chain to have an address on the remote chain for receiving funds and to eventually transfer those funds it controls into other accounts on that remote chain*
- *Teleporting: movement of an asset happens by destroying it on one side and creating a clone on the other side*
- *Reverse-Based Transfer: there may be two chains that want to nominate a third chain, where one includes a native asset that can be used as a reserve for that asset. Then, the derivative form of the asset on each of those chains would be fully backed, allowing the derivative asset to be exchanged for the underlying asset on the reserve chain backing it*

The XCM protocol is designed to be [10]:

- *Asynchronous: XCM messages in no way assume that the sender will be blocking on its completion*
- *Absolute: XCM messages are guaranteed to be delivered and interpreted accurately, in order, and in a timely fashion.*
- *Asymmetric: XCM messages do not have results. Any results must be separately communicated to the sender with an additional message.*
- *Agnostic: XCM makes no assumptions about the nature of the Consensus System between which messages are being passed.*

The actual message parsing between L0 and L1 chains is handled using queues. L0 can send messages to L1 chains by posting a message in the L1 chain's input queue. In a similar way, an L1 chain can post a message in the L0 input queue. This also enables the exchange of messages between L1 chains. It is the task of L0 to move transactions from the output queue of one L1 chain into the input queue of the destination L0 chain. In the L0 timeline, only the hash of the associated metadata is stored. The input/output queues must have a limited size to prevent spamming.

The Anatomy of an L1 inter-chain Interaction:

1. A smart contract that exists on L1 chain A will route a message to L1 chain B, where another smart contract is called to make a transfer of some assets within L1 chain B.
2. Charlie executes the smart contract on L1 chain A, which initiates a new cross-chain message for the destination of a smart contract on L1 chain B.
3. A validator of L1 chain A will place this new cross-chain message in the L0 inbound message queue, along with $message\_id$, $destination$, and a $timestamp$.
4. An L0 validator picks up the message and determines the destination L1 chain (using the $destination$ field), and places the message in the inbound message queue of L1 chain B.

5. An L1 validator of B will see this new message and add it for processing into the next block. While processing, the message will trigger smart contract execution on L1 chain B and complete the asset transfer as intended.
6. After finalizing the block on L1 chain B, its hash and $message\_id$ will be submitted to L0 for inclusion in the global timeline.
7. L0 will notify L1 chain A that the communication is completed and provide an optional result.

2.4 Smart Contracts

Substrate provides a Contracts pallet that allows the L1 runtime to deploy and execute WASM smart contracts. As described in the Substrate SDK documentation, the smart-contract code is stored once in a code cache and later retrievable via its hash. This means that multiple smart contracts can be instantiated from the same hash without replicating the code each time. When a smart contract is invoked, the associated code is retrieved and executed via the code hash. This call can modify the smart-contract account's storage entries, instantiate new smart contracts, or invoke other smart contracts.

Contract users must specify a gas limit with each call, as the smart contract's instructions require gas. Unused gas is refunded following the call, regardless of the outcome of the execution. When the gas limit is reached, all calls and state changes (including balance transfers) are reverted only at the contract level of the current call. For instance, if contract A calls B and B runs out of gas in the middle of the call, all of B's calls are reverted. Assuming contract A handles errors correctly, A's other calls and state changes continue to occur.

2.5 L1 Security

L1 chains have Proof of Stake (PoS) consensus and are assumed to be adversarial by default since they are permissionless untrusted networks of unknown participants. An L1 chain needs active validators and be alive for several epochs to be considered for inclusion in the L0 timeline and be able to consume L0 services.

In a multichain network, major security issues arise from cross-chain assets, because in PoS chains, attackers can hardly profit from a purely internal attack. On Mundis, if validators act maliciously in the L1 consensus process, anyone can challenge them by submitting fraud proofs to L0. Fraud Proofs present evidence that a state transition was incorrect. The main advantage of Fraud Proofs is that they are not needed for every state transition, but only when things supposedly break down. They require fewer computational resources and are a better fit for a scalability-constrained environment.

There are two types of fraud-proof corresponding to two kinds of malicious actions that can be challenged:

1. a group of L1 validators signed two different headers at the same height.
2. a group of L1 validators voted on a block that included invalid transactions.

The 1st type of fraud-proof can be verified by L0 directly. L0 tracks all public keys of L1 validators and uses the same public-key cryptography and curve with L1 chains. Once the challenge is verified, the slashing process is invoked automatically. Validators who signed the duplicate header will be slashed. The severity depends on the summed voting power of the faulty validators. If the voting power is 33% or more, 100% of the stake will be slashed. These

penalties are transferred to an on-chain L0 treasury. L0 will then halt the corrupted L1 chain, and its future would be dependent on a L0 governance decision.

The 2nd type of fraud-proof can't be verified directly by L0. So, once this type of fraud-proof is received, L0 pauses the staking/delegation operation and any cross-chain asset transfers into and out of the L1 chain. A governance process is then triggered on L0 to make a judgment and take corresponding actions. There is ongoing research to automate this process.

Another problem is data availability. While most systems rely on a data availability proof that which is complex and expensive, Mundis uses a challenge-response game. All L1 validators must continually observe L0, specifically the light client corresponding to its own L1 chain inside the L0, which acts as the Root-of-Trust for cross-chain asset transferring. A malicious group of validators could forge a header and update the light client in L0 but hide block content from honest validators. By doing that, the malicious group may steal cross-chain assets from the mainchain that are locked in the L0 bridge or transfer fake L1 assets to L0.
Suppose an honest L1 chain validators finds a newly committed block header in the chain's light client on L0 but doesn't have the corresponding block data. In this case, he would submit a query transmission to L0 expressing doubt. If he receives the block afterward, he would then withdraw the query. But if one L1 chain accumulates many queries on the same height, L0 will emit data availability challenges. In this situation, the header signers' responsibility is to submit a valid block that justifies the header. Should they fail to do so, their staking would be slashed. If the duration of this challenge-response game is significantly shorter than the unbonding period — and the total staking on the L1 caps the cross-chain assets — there is no chance for attackers to profit by hiding blocks.

3 Backbone security

Mundis is a bridging protocol where each L1 chain is sovereign and must maintain its own validator set and economic security. Since L1 chains are sovereign, the L0 chain is protected from L1 chain attacks.

L0 and L1 chains assume a byzantine adversarial model, where a minimum of $\frac{2}{3}n + 1$ L0 validators must be honest. Mundis uses a pBFT variant where the global timeline is used as a time source, reducing pBFT-specific messaging overhead and latency. If $\frac{2}{3}n + 1$ L0 validators are honest or haven't been compromised, Mundis can achieve secure consensus.

The L0 event timeline also records L1 block headers and basic Markle proofs needed for inter-L1 asset transfers. The timeline acts as an audit and additional verification mechanism for L1 chains. An L1 inter-chain transfer is secured by the ordered source and destination chain. Merkle proofs stored in the timeline that allows both ends of a transfer to be ordered and individually verifiable on each L1 chain.

C. The Multiverse

1 Layer 2 (The Construct)

Visitors and participants use a Human-Computer Interface (HCI) component to visualize and perform actions in the Multiverse. Mundis has several HCI components:

1. A desktop interface that provides the most complete and immersive experience

2. A mobile interface for working with Mundis using your smartphone
3. An Internet Browser with native Anima services support, wallet and an optimized WebGL rendering pipeline. This is meant to be a tool that will be widely adopted by Mundis users.
4. AR/VR devices in the future, once the technology becomes more mature and affordable.

2 Practicality

Practical technology

Mundis needs to be immediately accessible using current widespread technologies to reach a broad audience. While being less immersive than AR/VR Headsets, desktop and smartphone platforms have provided access to virtual realities for years. Their potential is proven by 3 billion currently active gamers.

Practical purpose

To become mainstream, the Metaverse needs to have a practical use for anyone, not just gamers, investors, and NFT artists.

We propose a virtual collaboration and socialization platform that can equip the user with creative communication tools. The untapped potential of the Multiverse lies in its ability to enhance interpersonal communication. Users gain access to a range of powerful dramatic tools and effects, enhancing communication and expression, giving the virtual medium new meaning and an important place in our everyday lives.

Practical UI

To enable wide-scale usability, Mundis introduces a UI that incorporates well-established gaming features but eliminates unnecessary or outdated conventions, like forcing avatars to walk to a certain point to accomplish a task that could have been solved with a click. Besides navigation and interaction in virtual spaces, Mundis will add new features to convey better and express information that includes avatar expression effects, P2P interactions, and social collaboration, allowing users to share information and interact in many ways.

The user can customize on-screen contextual information influenced by his surroundings. He can organize, search, prioritize, and integrate other apps to share text, images, watch videos, receive social media updates, make video calls, rate experiences, and provide feedback. Mundis will have deep social media integration to integrate social media content into the experience.

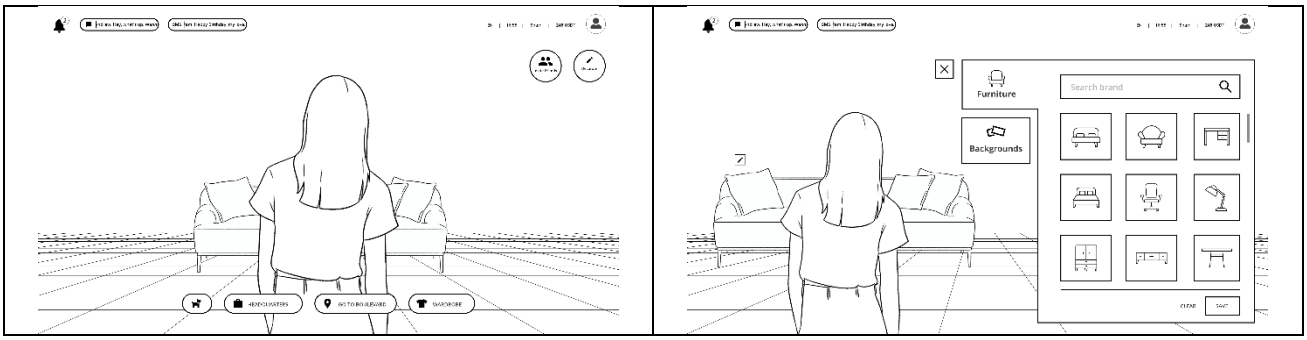
3 Shape and Structure

The Construct includes five main interconnected components where users can engage and perform meaningful actions:

- The Personal Space
- The Wardrobe Mall
- The Boulevard and side streets
- The Workplace with its Virtual HQ and Hub

3.1 Personal Space

It is the entry to Mundis, the start of the user journey. This fully customizable personal space will provide UI options and setups for work and a vast range of activities like social gatherings, games, concerts, etc.



The Personal Space will have the option to accommodate several live users. Their participation will not depend on a full-scale Mundis profile; any user can enter using a Guest avatar, only being required to install the app and supply a basic info set: email / username / password. Guests will have a limited UI but will fully participate in the experience.

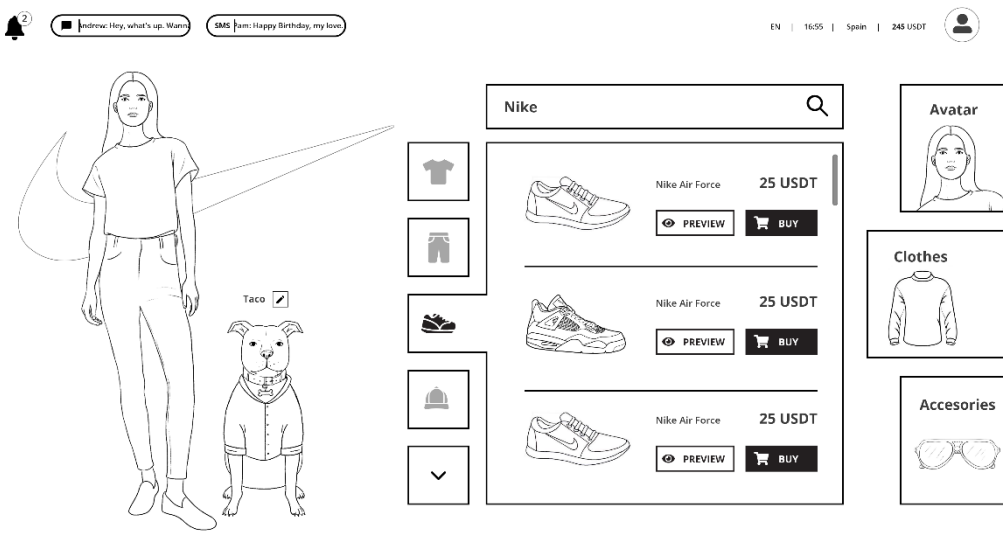
Users can set up their profile, add photos, videos, but also offer access to a non-live version of their Personal Space, a 3D extension of their profile that can be navigated and experienced by other users. Visiting users can experience media or original setups, either directly by entering the space or viewing a Friends Tunnel, an exploratory experience linking all friendly personal spaces into a walkable tunnel.

3.2 Wardrobe & the Wardrobe Mall

The Wardrobe will be used to create an "according to life" avatar, including detailed information like exact shoe size, skin tone, or hair complexion, and to create and edit your public avatar.

This avatar will not be limited to mirroring your exact form in the Metaverse and will provide many options for avatar personalization. However, to experience complete brand services - the "according to life" avatar will be mandatory to create. Still, it will be kept private at your discretion and only used for your benefit, giving you detailed and specific options on what info you share with brands about the interaction with their products.

In The Wardrobe, users can create the private "according to life" avatar with detailed information like exact shoe size, skin tone, or hair complexion, and their public avatar.



The mall's purpose is to allow users to try out and experience virtual brand products, ease In-Real-Life acquisition, or use dedicated commercial products from any Metaverse built for Mundis, right from their Wardrobe.

Products will be experienced here directly on your avatar through skins, acquired through a skin marketplace, clothing, accessories, or other NFTs you can use in Mundis.

3.3 Boulevard

Social and brand experiences will happen in a specific virtual interaction space you can access from your Mundis interface (and enter through your Personal Space). The Virtual Boulevard is a social space where you can explore and interact with other users or brands, play games, live meet, socialize, share information, and rate services, products, and brands.

3.3.1 Shape and urban style

The graphics of the Boulevard will attempt a highly realistic mirroring of life but will feature unique architecture and structures. The Boulevard itself will be shaped like a circle, with Side

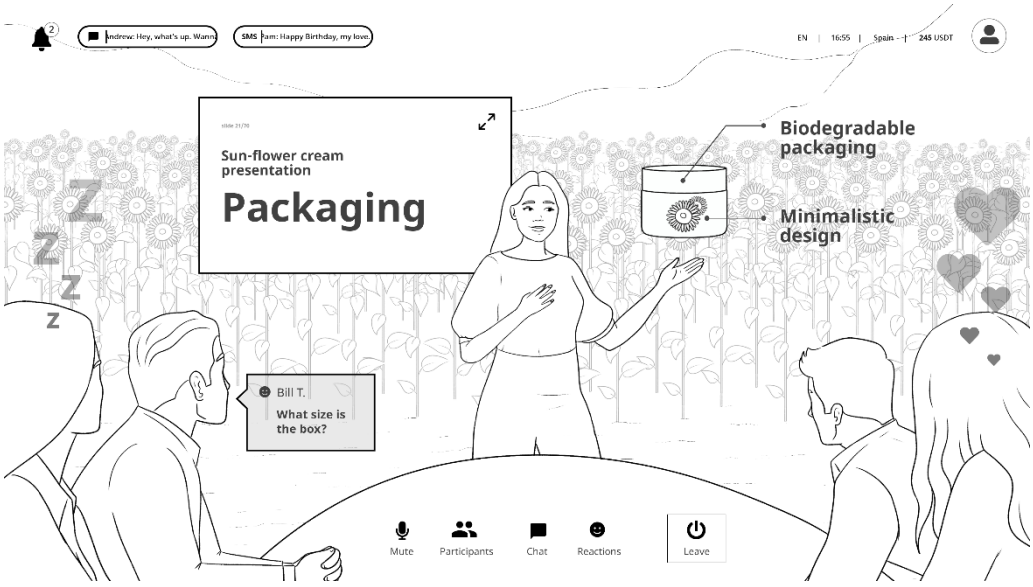
Streets developing on its exterior side. Side Street content will not be visible from the Boulevard, with users having to access the street to view its contents.

A cycle of day-night will be kept on the Boulevard. On half of the Boulevard will be day and the other night. It will take 24h for a complete day-night cycle, every spot on the Boulevard benefiting from both moods.

3.4 Workplace

We aim to offer a series of helpful presentation tools, brand manifestation spaces, and management structures that allow modern business collaboration. The Workplace is a virtual workplace for online meetings, relaying content, and presentations. It's a virtual environment accessible from your Personal Space, with unique fixed camera modes optimized for different scenarios: presentations, meetings, concerts, workshops, collaborative work, etc.

The available camera modes help participants to focus their attention on the organizer by viewing the complete Workplace scene together with text, video screens, effects, participants, and their feedback.



Instant access to your Workplace will be provided via invitation links. Anyone can join a Workplace through the invitation link as a guest or as a Citizen. The primary purpose is to allow users to create an immersive communication and presentation experience by enhancing data visualization, staging, virtual object, and image manipulation and using compelling novel video and text integration mechanisms.

A set of avatar and interface customizations, object inventory, setups, and actions are available by default, but 3rd party developers can contribute their content as NFTs.

3.4.1 The Virtual HQ

This is your new company office: a private, customizable space designed to help users inform their guests about their work or experience and showcase their venue. A unique visual manifestation space can be created with a personalized interface and action set, an adaptable display space, a programable experience tool, and a meeting space that can accommodate various setups and scenery.

3.4.2 The Hub

A new way to enhance daily work experience in your company, do employee training and development, motivate your team, or have some fun.

The Hub is an organizational add-on to the Virtual HQ, offering multiple setups and controls. As a result, organizations can translate valuable processes in the virtual space, like brainstorming, presentations, project follow-up, and teambuilding. It will also allow the development of a healthy and engaging organizational culture using gamification elements and progressive accumulation.

4 Key Activities

Mundis will be designed to accommodate specific activities, both structurally and from a UI/UX perspective.

4.1 Social Interaction

Mundis is a venue for peer-to-peer information exchange, socialization, and social aggregation. Users will be able to present a personal profile and a feed, check out other profiles, interact, and form and join groups.

Friend lists, profiles, feeds, groups, and other social media tools will be accessible, offering UI integrated classic search and visualization options.

Meeting people and general interaction in Mundis will start in your Personal Space, where you can invite your friends and trade NFTs. It will continue on the Boulevard and its side streets, where you will be able to play games or enjoy other experiences by visiting side buildings.

Mundis includes engaging communication tools, character action sets, and controls to enhance socialization in all virtual spaces. This UI will focus on virtual communication, forgoing all unnecessary tropes of the gaming paradigm (e.g., the need to perform an unnecessary movement to reach a certain marker with your avatar to accomplish a certain task), and minimize efforts to allow more user expression.

The user can customize on-screen contextual information about what he sees. He will organize, search, prioritize, and integrate apps to watch videos, receive social media updates, perform video calls, view documents, rate experiences, provide feedback, and integrate content from social media platforms.

The non-comprehensive list of interaction capabilities is designed to maximize social interaction via a set of options:

- Communication: live audio and video capabilities, live texting, live document sharing, and visualization
- Expressing emotions: avatar mood add-ons, personalized avatar unique gestures & effects
- Rating: an engaging service/experience rating system
- 3rd party content display: the possibility to showcase your social media posts, stories, personal information, photos & videos. Options to showcase your game stats, game sessions, or live play

The UI will be adapted for each specific environment. For example, users have some available options on The Boulevard, while other options will be enabled or disabled in their

Personal Spaces and Workplace. The set of available options for Boulevard and Personal Space interaction includes, but is not limited to:

- Speech bubbles: the avatar will display speech bubbles to add to an ongoing voice/video conversation.
- Mood indicator: a voluntary setup that you share if you are bored, happy, sad, horny, or whatever you want, and it will hover by your avatar name above your head.
- On scene avatar entry and exit effects: each avatar will configure his entry and exit effects, going in and out of a Personal/Meeting Space. From an elegant "teleportation" effect to less conventional "Kramer" entries, users will be able to extend their personality through various styles of effects.
- Avatar emotion effects: ranging through all emoticons and focusing on the most used. The users/avatars should be able to use emoticons in a variety of ways, from embodying an emoticon (displaying a smiling emoticon instead of your avatar head) to handing out a little card containing an emoticon (and other cool Emoji reveals that can add style to your feedback)
- 3D text as statements: avatars will display text either by holding it or having it hover above their head.
- Comparison module: Instantly draw out temporary objects displaying animals, objects, and more from a comparison library.
- A fun voting system: Where users can use stars, grades, colors to:
    - vote on multiple options and add feedback to each
    - vote on a single thing and show the level you want to support it

4.1.1 Boulevard Interaction

The Mundis UI will offer a set of tools to create extensive ways for avatars to interact in the virtual world. Interaction with space will also provide specific options that ease socialization, help create mood and enhance shared experiences.

Different modes for viewing your avatar on the Boulevard will be available (1st person POV, close and far 3rd person).

Avatars will be able to cruise The Boulevard and enter sidewalk apps in the form of buildings where they can solve brand or user quests for financial or experiential gains, use special places to interact in unique ways, visit business HQs, or step into dedicated brand created worlds/spaces.

The Boulevard will be available both as a walk-through experience and as an easily navigable space via the user interface. In addition, different ways to walk through the Boulevard will be made available, including driving, biking, or flying, each with its separate UI interface and POV.

The walk-through experience UI can be customized to contain a detailed information feed that the user can closely adjust to his preferences. This feed will be projected over the visible field and will also influence your personal interface and options, allowing you to also visualize live feeds, videos, text, messages or to share screen.

4.1.2 Public presence

Users will be able to access the Boulevard as a shared experience and to temporarily pin their Personal Space on a designated free part of the Boulevard for 24h. After this period, the Personal Space pod will be automatically removed. The appearance of the Personal Space on the Boulevard can be made visible to anyone. This appearance will also use skins. These skins can feature messages, videos, or effects.

If no space is available for pinning, side streets can also be used for temporary pinning or the Non-Stop Personal Space pinning spaces at the end of any side street. Any number of PS can dock here with no time limit. As the side street grows and features more stores, the Non-Stop area will be pushed away to make room. These Non-Stop areas will provide ample space for communities to grow and develop; the size of these areas will also influence the value of the side-street spaces.

4.2 Collaboration

The Mundis UI/UX is adapted for professional collaboration, events, workshops, and team communication via The Workspace, HQ, and Virtual Hub.

The Workspace can accommodate large meetings where users can invite their colleagues, fans, and friends to collaborate, present their ideas, showcase a product or service, or even hold a concert or public speech/workshop.

The Scene Editor can be easily used to construct and edit full virtual experiences for your guests or team, including timed appearances, effects, or other mechanisms.

The set of available options for Workplace interaction and presentation includes, but is not limited to:

- Collaborative Whiteboard: A collaborative tool that enables the participants to a meeting to add text, images or to draw on a highly visible whiteboard
- Dramatic stage manipulation: Instantly change the setup/scenery to add a dramatic effect. Meet in a generic presentation room but switch to a forest setting when you talk about sustainability or the seaside when you want to sell a soft drink. Dim lights progressively, shrink or enlarge the scene, add weather effects, use light spots
- Conceptual data models and Formulas: Where you can quickly form floating formulas or data models and use shapes, text, images, or video instead of numbers, helpful in attempting to explain complicated concepts or models
- Collaborative 3d object manipulation: Complex 3D objects can be moved around or edited by all meeting participants
- Evolving elements: Intended as presentation props, evolving elements can be displayed near the presenter to create a more evocative mood: an erupting volcano, a growing tree, a burning bridge, aging, a depleting glass, a fast-revolving sun on the sky, etc.
- Unveiling: Add various revealing effects to 3d text, images, documents, or video. Either as a typical unveiling or to suggest a Tip of the Iceberg effect, where the user starts with a small thing, but reveals its true size to the audience
- Bots: Bots under various forms can join your presentation, supporting your presentation flow. They can be set up before the presentation or summoned in the presentation ad-hoc. Their actions can be programmed to follow specific markers with various actions, and their movement patterns will be automated by an AI. Specific bots with special programming can also be summoned, like Einstein or Lara Croft "lookalikes."

- Time Management (Measuring, Censoring): Limit meetings or personal speech time with hovering timers, personal time-bars, room shrinking effects
- Randomizing effects: Using different randomizing systems and effects to assign teams, giveaway assignments, or designate jobs to multiple people. Systems include a fortune wheel, a rock/paper/scissors game, a spinning bottle, etc.

4.2.1 Headquarters (HQ)

The Headquarters (HQ) serves as an actual office for your organization in the Mundis Multiverse. Authorized users will be able to edit this space to fit presentations or meetings, showcase the company portfolio, and set up experiences or games for any guest to take part in.

Specific special effects or avatar actions will be available only in HQ mode. These can be custom-made or acquired by the company as NFT's from the Marketplace.

4.2.2 Business Processes

Built as an internal organizational resource, the Hub supports project management and team communication with special workflow and reporting tools that incentivize collaboration and productivity.

Custom group games and activities can be set up and even personalized for each team member. These set-up-controlled experiences can serve as briefing support, onboarding guides, or fun teambuilding exercises.

Activities of team members can be incentivized and sustained by a company token, gamification quests and tasks, and various play-to-earn or work-to-earn systems.

4.3 Gaming

Gaming is an integral part of the Mundis experience, and all interaction with the platform will be gamified to an extent. On-platform gamification will reward the evolution of your avatar, the types of interaction users get involved in, and the feedback you get from other users. A brand experience can also be gamified, allowing parallel quests, the gathering of experience, or other reward systems to motivate users.

An array of Play-to-earn and Create-to-earn systems and options will be provided to any user.

4.3.1 User experience

Your personal evolution in the Multiverse, including changes on your avatar, editing your Personal Space, trading, using Business Mode, cruising The Boulevard, or interacting with users and Brands will be motivated by a series of quests and tasks, with prizes in crypto, useful NFTs (skins or items) or communication effects.

A user rating system will also change in value depending on your actions and the feedback you get from other users.

Mundis will offer a series of games to familiarize its users with the Multiverse, help them master new skills like trading or creating NFTs or just offer a fun experience. These games will have separate systems with their own rewards.

4.3.2 Brand experience

Upon acquiring space on the Boulevard or side streets, brands will be able to offer their own play-to-earn experiences to Mundis users, offering NFTs, tokens, or other rewards Completely separate ecosystems can be integrated into the Mundis metaverse, with their own currency and rules.

Users will be able to access them from the Boulevard or directly from the UI.

4.4 Investment

Users can invest in assets like venues, locations on the Boulevard, or NFTs. Each new venue can provide its own separate opportunities for investment and trading

The Launchpad is an external platform where you can support various Token projects proposed for Mundis or participate in auctions where users can bid for locations on The Boulevard to use or rent.

A peer-to-peer NFT marketplace acts as a decentralized exchange, allowing users to trade assets.

Creators can sell their content as NFTs in the Marketplace. In addition, NFT sellers can receive payment for their creations in any supported Mundis token.

Mundis allows creators to launch a custom storefront, like Shopify or WordPress. It provides a graphical interface to on-chain NFT system programs for creators, buyers, and curators of NFTs. In addition, the design and layout of storefronts can be customized either as a permanent store or a temporary place for a specific auction or collection.

Developers can build custom worlds, chains, and technology projects and advertise them on the Launchpad to gather L1 validator support and initial capital.

5 Access

Gaining access to the Mundis boulevard or its side streets is a two-step process. First, the entity needs to purchase a venue that can be placed on the Boulevard. This venue is an NFT and can be:

- a bespoke building created by Mundis for special partners
- a standard building with preconfigured branding options (e.g., dedicated info spaces, logos, or other info devices)

Purchasing the venue will enable the option to bid for a specific location on the Boulevard. Mundis holds auctions for all boulevard locations on a fixed schedule (e.g., trimestral) dictated by network consensus rules.

Boulevard auctions happen on-chain, so anyone that holds a venue NFT is free to submit their bid. Bids contain two pieces of information: the location on the Boulevard and the duration of the lease slot (e.g., max one year). Auction winners must lock up their bid in a bond for the duration of the lease slot at the beginning of the auction.

6 AR/VR Devices

AR/VR experiences can be created in Mundis the same way as other game-like experiences. Mundis does not restrict what can be developed, it just sets some standards on how to develop and how to integrate your new experience in the Multiverse.

7 Citizens

Mundis users become Citizens by getting an identity in the Multiverse. Some Anima services are restricted only to Citizens (i.e., LaunchPad, Location, Marketplace), but an L1 chain can also make its services available only to Citizens if the builder of the L1 chain chooses to do so.

A Citizen can have an active role in the Multiverse or a specific L1 chain. In addition, he can benefit from being involved in one or more L1 chains during specific events (e.g., an airdrop when a new L1 chain is bootstrapped).

Users will have a public identity card in Mundis that is a Master Edition NFT uniquely identified by an address. The identity card can have public and private attributes.

Public attributes enhance a user's Mundis profile with the following information:

- non-unique alias (e.g., megatr0n)
- social media handles (e.g., @megatr0n for Titter, megatr0n#5431 for Discord etc.)
- a small rectangle avatar image (50x50 24bit/pixel JPEG, stored in IPFS)

Private attributes are encrypted using the wallet's private key. Their hash will be visible on the public identity card, but only the owner can decrypt them.

Users can share the identity card with interested parties as Print NFTs. This enables the owner to keep track of all issued Prints for security purposes.

Under special circumstances, a user can share private attributes with another party using a special authorization flow. A requested first signs a message requesting the user's private details. Then, the user verifies the signature and issues a special Print NFT with the private attributes re-encrypted using the requester's public key. This way, only the requester can see the private information.

D. The Ecosystem

1 NFTs

In Mundis, Non-fungible assets (NFTs) have some important distinctions to fungible assets (tokens). Mundis NFTs are described in MDS-721 and must be:

- Unique: each NFT is uniquely identified by an L0/L1 address in Mundis and optionally an Anima storage hash for its resources.
- Ownable: an NFT can be owned by a single citizen.
- Immutable: once issued, an NFT cannot be destroyed or altered.
- Transferrable: the ownership of an NFT can be transferred between citizens and an L1 chain.
- Displayable: an NFT can be optionally displayed in Layer 2 if it's MDS-722 compatible.
- Categorizable: NFTs must adhere to an ontology in Mundis (i.e., digital art, avatar, car, shoes, etc.) defined in the MDS-723 standard.

Mundis NFTs will extend the Metaplex Protocol [11] to support:

- creating/minting NFTs

- auctions for primary/secondary sales
- visualizing NFTs in a standard way across wallets and applications
- creating storefronts
- royalties for collaborating on creating content

The mission of Mundis is to standardize the NFT protocol to a greater extent, allowing anyone to easily mint, display and monetize their creations in the Multiverse and all associated Metaverses.

1.1 Standards

To accommodate many NFT use cases, Mundis implements the following types of NFTs:

a) Master NFT

A master NFT is a non-fungible token and associated metadata that allows creators to control the provenance of prints created from the master NFT. Rights to create prints can be tokenized, the owner of the master NFT can distribute additional tokens to allow users to create prints. Additionally, the creator can set a max supply for the master NFT, just like a regular mint on Mundis, with the main difference being that each print is a numbered NFT created from it. A notable and desirable effect of master NFTs is that as prints are sold, the artwork will remain visible in the artist's wallet as a master NFT, while prints NFTs appear in the purchaser's wallets.

b) Print NFT

A print NFT is a copy of a master NFT. Each print has a number associated with it to uniquely identify it. Prints can be created as a part of an auction on the Mundis NFT Marketplace but can also be minted manually by the creator.

c) Standard NFT

A normal NFT (like a Master NFT) that is a non-fungible token and associated metadata but lacks rights to print. It can be a one-of-a-kind artwork that when sold, is transferred from the artist's wallet to the purchaser's wallet.

Additional mechanisms for revocation, time, use, and data-based expiry, explosion, and selective/non-transferability are available in Mundis to support Metaverse use-cases like issuance, ownership, and management of programmatically governed tokens to facilitate rental agreements, loans, tickets, domain names, gaming items, certifications, credentials, badges, and much more.

1.1.1 Transferability

NFTs in Mundis support three degrees of transferability:

- Frozen
- Permissioned transfer
- Permissionless transfer

If the issuer chooses to make transferability permissioned, he must choose a transfer authority to approve transfers. A token holder can then only transfer ownership by invoking the transfer method which will require a signature from the transfer authority. If the token issuer choses to make the token freely transferable, a transfer authority is not required.

In all circumstances, a signature from the transfer receiver is required to correctly invoke the transfer mechanism. This is due to the fact that the transfer method reassigns ownership of the current owner's token account, which by default removes the current delegate. Before the token account is re-frozen, a signature from the new owner is necessary to reset the token manager as a delegate.

1.1.2 Payment Settlement & Extension

NFT issuers can optionally chose to collect funds throughout a NFT lifecycle and then pay out these funds either upfront or upon invalidation. When choosing to accept funds, the issuer must specify the payment mint and add a paid-claim-approver with the amount specified to ensure it must be paid to claim the token.

It's also feasible to provide invalidators extensions so that they can accept payment and avoid invalidation. It is up to the issuer to use a payment-accepting invalidator and then establish an extension price within it. Allowing the invalidator to be extended proportionally to the initial payment and length is an example extension schema.

1.1.3 Expiration and Invalidation

When constructing a token-manager, one or more invalidators might be assigned as the only public keys that can perform an invalidation. A permissionless call can invalidate the token in the case of time or usage-based invalidation. After that, all appropriate payments are made, and the product's lifecycle is complete.

Mundis has the following built-in invalidation mechanisms:

1. Time: the issuer can set a duration at issuance or claiming. The token can be invalidated after the expiration date.
2. Usage: the issuer can set a maximum number of usages. Token holders declare usages by signing transactions. If the number of usages exceeds the set maximum usages, the token can be invalidated.
3. Manual revocation: the issuer can specify a revoke authority that can invalidate the certificate anytime.

Mundis also supports different invalidation types. These specify what happens with the token after an invalidation took place:

1. Return: after invalidation, the token will be returned to the issuer
2. Invalidate: after invalidation, the token will remain marked as invalid and can no longer be used
3. Release: after invalidation the token-manager will be deleted and tokens are released

1.2 Auctions

The NFT marketplace will supports four types of auctions derived from English auctions:

1. Single Item: can be used to sell Standard NFTs and re-sell Print NFTs, as well as the sale of Master NFTs themselves (and the associated printing rights) if the artist so wishes.
2. Open Auction: requires the offering of a Master NFT that specifically has no fixed supply. The auction will only create Print NFTs for bidders. Each bidder is guaranteed to get a unique print. (there are no "winners" in this type of auction). An open auction can:
    - have a pre-set fixed price
    - have the bid price set to the last offer
    - have no price (be free)
3. Limited Auction: a Master NFT may be provided to the auction with a number of copies that represent the number of winner places. For each winner, a Print NFT will be minted and awarded to the winning bidder of that place. For example, the first-place winner will win Print #1; the second-place winner Print #2, etc.

4. Tiered Auction: a mix of the previous auction types as winning placements. For instance, the first-place winner could win a Print of Limited Edition NFT A, while the second-place winner could win a Standard NFT, etc. In addition, participants who did not win could get a Participation Print NFT from the Master NFT.

1.3 Royalties

Each NFT can be minted with configurable royalty payments that are then sent automatically back to the original creators whenever an artwork is resold on the NFT marketplace in the future.

1.4 Marketplace

Creators can sell their content as NFTs in the NFT marketplace. The Marketplace will be available in native Layer 4 apps and as a dApp accessible through any internet browser. NFT sellers have the option to receive payment for their creations in any supported Mundis token.

2 Wallet

Payments and earnings in Mundis are made using ecosystem tokens. The Metaverse app will include a non-custodial wallet to help users keep track and organize their funds, income, and expenses.

Through the wallet, users can store, stake, swap and make transfers using Mundis ecosystem tokens. The wallet will also store NFTs and is available natively in desktop, mobile apps, and the Layer 2 and has built-in asset transfer capabilities between L0/L1 chains and external blockchains. Citizens can teleport NFTs and tokens from other blockchains into Mundis.

3 LaunchPad

The LaunchPad is the central point for exploring and investing in new projects, ideas, and auctions in the Mundis ecosystem.

3.1 Token Projects

Projects and ideas that require community involvement or financing should register on the Launchpad to describe their idea. A project may ask community support for L1 validator seats and/or crowdfunding in exchange for the project's tokens. Each project will define its own rules, rewards, vesting schedule and token release terms.

Anyone can request to publish their project on the Launchpad. However, to prevent malicious behavior, projects need to be vetted by L0 validators before having access to investors.

Access to Launchpad projects will be based on a "lottery format" to ensure fair access to all investors. Users can purchase up to 5 lottery tickets prior to a sale. Specifically, the exact number of tickets a user is given is determined based on their Mundis account balance "over a 20-day period." If a user maintains a sufficient account balance for said period "leading up to the day of the lottery," then they will reportedly be awarded "a maximum of 5 tickets per eligible account.".

The Launchpad will perform a transparent, randomized selection of lottery tickets 24 hours before the sale and users need to confirm how many tickets they want to purchase. Once a user confirms the number of tickets, it commits to purchase the number of tokens corresponding to number of winning tickets."

Each new project listed on the Launchpad will reveal the maximum number of potential lottery ticket winners. Moreover, the specific "allocation amount" associated with each "winning ticket" will be announced "ahead of time." (e.g., 10,000 potential winning tickets, each representing 500 USD worth of tokens).

3.2 Auctions

Besides token projects, Mundis will also host regular auctions for places on the Boulevard. Auctions use the candle mechanism to randomize the exact moment the winner of the auction is determined. Participants have a pre-set amount of time to place bids and, to ensure fair bidding, the time when the winner is determined is randomly decided on and retroactively applied to the highest bidders.

Historically, candle auctions were used in the 16th century as a way to sell ships. At the beginning of an auction the candle is lit and the highest bid at the time the candle runs out wins the ship. So, while there is an official end to the auction, a candle auction refers to the random spot in time that will determine the winner. This random spot is always before the official end of the auction.

In addition to direct participation by a single bidding entity, Mundis enables anyone to crowdsource support for their idea using the LaunchPad. In this approach, Mundis token holders can bond their tokens to signal support for a particular project in exchange for project-related benefits (e.g., airdrops, discounts, yield etc.) If the project wins, the bonded tokens are locked by the network for the duration of the lease, after which the tokens are unlocked and may be reclaimed by those who contributed.

If the auction is not won, the bonded funds are returned to their owners.

4 DEX

A decentralized exchange (or DEX) is a peer-to-peer marketplace where transactions occur directly between crypto traders. Mundis provides a DEX for traders that wish to trade ecosystem tokens.

5 Development

Mundis relies on developers to expand its Multiverse with:

- Content items packaged as NFTs for:
  - buildings
  - avatar mood add-ons, gestures, effects
  - avatar clothing & accessories
  - personal space layouts, scenery, items
  - workplace scenery, items

  Content items will be designed in an open, interchangeable format like glTF (GL Transmission Format) to be easily displayed on web, mobile and 3rd party apps.

- Custom experiences to be used in buildings, private space, and the Workplace (scenes, interactions). These experiences are usually guided interactions for a user in a spatially-restricted environment (e.g., visit a museum, an art gallery, a shop-like experience, a quest, etc.).

- Metaverses for complex scenarios like games, exhibitions, VR environments. These worlds are built using the Mundis world builder app and contain their own 3d environment, rules, assets, interactions, and economy. A custom-built world can have its own L1 side-chain for governance, tokens, and economic rules.

- New technologies for extending Mundis with additional capabilities (data oracles, DeFI, GameFi, payment protocols, etc.). These developments will always be done using L1 chains and are usually showcased in the Launchpad. Mundis provides its own Substrate-compatible SDK for developing L1 chains.

- Existing apps and games. Integrating existing systems in the Multiverse should be easy. Any application or game creator can add integrate their Metaverse into Mundis using the provided Mundis SDK. Adding NFTs or enabling token payments inside your app should be trivial, and this will rapidly push the adoption of Mundis. A simple onboarding process and easy integration options are the ingredients for global adoption.

E. Standards

A main objective of Mundis is to bring uniformity to the Metaverse space by introducing universal standards for development, integration, payments, 3D objects, rendering and more.

The goal is to promote interoperability and usability not just to the Mundis ecosystem, but to future Metaverse projects.

Standard	Description
MDS-721	Ownership of non-fungible tokens
MDS-722	2D/3D Display standard for assets
MDS-723	Non-fungible token ontology
MDS-724	Non-fungible token royalties
MDS-137	Mundis Name Service

F. References

[1] Polkadot and Cosmos · Polkadot Wiki. Available: https://wiki.polkadot.network/docs/learn-comparisons-cosmos
[2] Solana Whitepaper. Available: https://solana.com/solana-whitepaper.pdf
[3] Bridges · Guide. Available: https://guide.kusama.network/docs/learn-bridges/
[4] XCLAIM: Trustless, Interoperable, Cryptocurrency-Backed Assets. https://eprint.iacr.org/2018/643.pdf
[5] IPFS - Content Addressed, Versioned, P2P File System (DRAFT 3). https://block.academy/researches/ipfs.draft3.pdf
[6] Shinobi Systems' Primer Available: https://www.shinobi-systems.com/primer.html
[7] Practical Byzantine Fault Tolerance, Miguel Castro and Barbara Liskov Available: https://pmg.csail.mit.edu/papers/osdi99.pdf
[8] Tower BFT: Solana's High-Performance Implementation of PBFT Available: https://medium.com/solana-labs/tower-bft-solanas-high-performance-implementation-of-pbft-464725911e79
[9] Substrate Developers Guide v3 Available: https://docs.substrate.io/v3/runtime/frame/
[10] Polkadot Cross-Consensus Message (XCM) Format Available: https://github.com/paritytech/xcm-format
[11] Official Metaplex Docs and Guide: https://github.com/metaplex-foundation/metaplex
[12] Cardinal protocol: https://cardinal.so