SKY PROTOCOL

DATA AVAILABILITY NETWORK

WHITEPAPER Version 2.1

This page is intentionally left blank.

Abstract

Sky Protocol is pioneering a Layer 2 solution tailored for Cardano, dedicated to revolutionizing data availability. Sky Protocol unbundles four essential aspects of blockchains—consensus, validation, data availability and bridging—so that each Decentralized Application (DApp) can use a combination of those aspects that best fits its needs. Sky Protocol's modular design enables better throughput, latency, privacy, safety, censorship-resistance and cost compared to existing blockchain technologies. There are of course tradeoffs involved between some of these qualities; but Sky Protocol enables each DApp to enjoy the qualities its users care about that existing one-size-fits-none systems can't provide, without having to pay a dear price for those qualities they don't care about that those systems force upon them. Future versions of Sky Protocol will support not only Cardano, but all blockchains that matter, including the ability for each DApp to directly bridge between blockchains; and they will also provide modular validation and consensus services as well as data availability. In the end, Sky Protocol aims at drastically reducing the cost and complexity of building DApps—so much that within a decade, the default would be for any new application to be decentralized rather than centralized. This abstract offers a comprehensive overview of Sky Protocol's architecture, functionalities, and transformative potential.

Disclaimer

The contents of this white paper are subject to potential revisions, may entail unforeseen risks, and could lead to new discoveries necessitating a reassessment of our initial assumptions. The Sky Protocol team retains the right to amend the white paper and project specifications for any reason. This white paper is intended to complement the code deployed by Sky Protocol. It's essential to underscore that the definitive source of accuracy and reliability is always the code itself. The technical information presented in this paper should not be construed as an exhaustive list of features, and it's worth noting that features may also be subject to removal.

Contents

1. Introduction

1.1. Summary

Today, Scalability—the ability to process a large number of transactions in a timely fashion—is a limiting factor for Decentralized Applications ("DApps"), to the point where most applications resort to a centralized operator. Most (all?) "Layer 2" solutions to scale the handling of decentralized assets from "Layer 1" blockchains sacrifice or compromise Decentralization one way or the other. The Cardano blockchain in particular, though promising by its robust design and healthy fundamentals, is lacking in scalability solutions, even centralized ones.

Sky Protocol is a modular solution to Layer 2 scalability that will launch this year (2024) on the Cardano blockchain, and on other blockchains afterwards. By going back to first principles, Sky Protocol can sport a modular design that pushes the boundaries of the tradeoffs involved; it thus allows each DApp to enjoy the capabilities and capacities it needs, without sacrificing decentralization, yet without burdening the necessary resource-limited Layer 1. Each DApp will have a DApp-appropriate level of throughput, latency to finality, smart contract capability, privacy, censorship-resistance, decentralization, cost-effectiveness, ability to interact with outside systems, etc. Sky Protocol will thus offer the Cardano community a rapid and secure DeFi experience.

Our choice of Cardano as a first blockchain to support is strategic for both Sky Protocol and Cardano: Cardano is unique both in (a) being designed for robustness and smart contracts, as well as (b) having a governance structure both decentralized yet able to deal with centralized institutions—thus providing the right environment for Sky Protocol to demonstrate its capabilities. Yet, Cardano also has many opportunities for improvements, and Scalability in particular is a topic where it can best benefit from the contributions of Sky Protocol.

We'll explain these claims in more detail in this introductory section, then will give more details on the design of Sky Protocol.

1.2. The Problem: Scaling Decentralized Applications

1.2.1. Centralized Applications

Today, most Software Applications are Centralized: they are run under the control of a central operator, whose code is secret, and who will spy on users, use ads to divert their attention, control what they are able to see, censor people or information they don't like, and otherwise serve the interests of the mighty to the detriment of the users.

Banks and credit card companies, even in so-called “Democratic”, “First-World” “Countries”, have been systematically refusing to process transactions for people who have the disfavor of the regime; not just sex workers and drug traders, not only gun dealers, but also journalists who will denounce corruption of officials instead of spreading their propaganda, truckers who peacefully protest their totalitarian lockdowns, or even friends and family of protestors who try to support them—sometimes even locking or seizing the funds of the victims. Meanwhile, they are using their monopoly powers to conduct massive inflation, robbing the value of holdings of every productive citizen to transfer wealth to the pillars of the regime, as well as to fund massive wars, at unprecedented scale.

Social Networks and Search Engines, even in the same “Democratic First-World Countries”, have been massively censoring points of view that threaten the Establishment while pushing regime propaganda. Their so-called “fact checkers” admittedly being pushers of the official ideology.

The regimes often justify their massive spy networks in the name of saving the world from child abusers and terrorists, which is belied by numerous known cases where they do little to nothing against members of well-known child abuse rings (such as clients of Jeffrey Epstein), and have conducted massive funding of terrorist groups (such as Al Qaeda in Syria). But whatever the official justifications for central control, the actual result is that any centralized software application is soon captured by the Establishment to become part of a massive apparatus for the surveillance and brainwashing of the population.

Still, Centralized Applications (CApps) were historically discovered first, because they are simpler to build and run, and the problems with central control were secondary to having an application run at all to begin with.

1.2.2. Decentralized Applications

There is an alternative: Decentralized Applications (DApps), i.e. applications wherein no single participant or small group of participants holds the power (a) to control inflation to deal resources to themselves (“inflation resistance”), (b) to censor transactions they don’t like (“censorship resistance”), and sometimes even (c) to see the data of transactions they are not parties to determine whether they like them or not (“transaction privacy”).

The first successful such application was Bitcoin, the peer-to-peer electronic cash system, that uses the “coins” managed by its ledger as incentives for people to honestly participate in the process of validating the “blocks” of transaction data that constitute the ledger itself, using the innovative “Proof of Work” mechanism to ensure consensus on “the” state of the ledger. Despite all its limitations and inefficiencies, Bitcoin proved that it was possible to create a monetary system free from central control, censorship or inflation, and remains a fast growing success to this day.

Since then, many Decentralized Applications have been designed; they usually reprise some variant of the “blockchain” data structure of Bitcoin (though some of them have neither blocks nor chains, but instead sequences of transactions or DAGs), or build additional structures on top of one that does. [TODO: give a non-exhaustive but diverse list of notable such systems—MasterCoin, Namecoin, Ethereum, Uniswap, xDAI, ENS, Solana, Optimism, Arbitrum, Celestia, etc.]

These systems hold the promise of a world where users are not held hostage by their software service providers anymore.

However, this promise is still largely unfulfilled to this day. DApps do help a lot of people evade the restrictions from the centralized monopolies in the more oppressive countries. Yet overall, most blockchains seem to be mostly vehicles for speculation, and not much in terms of actual platforms for DApps.

1.2.3. Technical Hurdles to Decentralization

Today, writing a DApp is extremely difficult, running one is expensive, and existing systems can only handle so much volume of transactions.

For instance, as of early June 2024, the Bitcoin network can handle about 2800 simple transactions every block (1MB limit, about 374 vB per simple transaction), one block every 10 minutes, for an average peak throughput of about 4.6 transactions per second (“tps”), with a cost of about $0.81 to get a first confirmation within 60 minutes for a typical simple transaction. Its rival, Ethereum can handle about 714 transactions (15M gas limit, 21000 minimum gas per transaction) per block, one block every 13s, for an average peak throughput of about 55 tps, at a cost of about $0.73 to get a first confirmation within 3 minutes for a typical simple transaction. Neither the throughput limits nor the transaction costs make either system acceptable as a replacement for centralized systems, that can typically handle many tens of thousands of transactions per second, and beyond.

Some newer blockchains, such as Solana, TON or Tezos, have recently advertised performance that rivals with centralized credit card processing systems. This is great progress, but even then, this is still far from what is needed to move all human software to DApps; and then there is the problem of bridging those systems with those other systems, where the valuable tokens are, or where the useful action happens.·

1.2.4. Economic Limitations to Solving Issues

Many DApps seek to address scalability issues with blockchain. However, in most cases, these DApps build a protocol that is specific to a single DApp; then they need to raise capital around a new token for a new economic validation network, from scratch, but the capital raised will only be used to guarantee that single DApp, which is not efficient.

Now, some DApps aim at becoming a universal DApp that can be programmed to emulate any other DApp, so its economic validation network can be universal. Bitcoin initially hoped to be that but quickly stopped trying, and its technology has consistently failed to evolve, in a massive governance failure; it remains the biggest network, with the hope that it will eventually adopt change before it becomes irrelevant. Ethereum actively tried to be the universal, programmable DApp; but the enormous success of its great talent comes with both inertia and self-indulgent blinders that limit the dimensions of design that it may evolve in; in particular, its . There are a few contenders, though they lag behind in market- and mind- share.

One interesting new entrant in the bid for universality is EigenLayer, that has a great story why its token can be more universal than the previous ones in a way; though it remains to be seen in practice how their token can both diversify the tasks it protects yet not run into unstable or insecure usage patterns.

Bitcoin has the greatest network effect; but, forks, etc. Ethereum is a strong contender with a somewhat functional governance structure, and a lot of mindshare; but its Account model strongly limits opportunities for parallelization and scalability, and inertia will make a fix very hard.

1.3. The Solution: Sky Protocol

1.3.1 The Challenge

Now what if we could radically improve DApp technology, so it becomes easier and cheaper to develop DApps than Centralized Applications (CApps)? What if DApps could be made to scale just as well as CApps, at a running cost lower than that of CApps, while providing better guarantees for the end-users? We argue this is possible, but requires going back to first principles to understand how DApps and CApps work.

1.3.2. Value Proposition

Sky Protocol proposes a modular architecture for building DApps that:

1. Radically simplifies the work required to build DApps.
2. Allows DApps to seamlessly scale as their usage grows.
3. Will eventually support bridging to all blockchains.

The end-goal is that eventually, most people writing new Apps in the future will write DApps by default, not CApps, and most DApps will use Sky Protocol.

Sky Protocol will simplify the work required to build DApps by offering a modular architecture (see section 2 below), wherein developers can mix-and-match components for consensus, validation, availability, and bridging. Moreover, Sky Protocol will enable developers to metaprogram all the parts of the DApp (on-chain for every chain at stake plus off-chain for every participant) from a single high-level specification, whereas today you have to manually and redundantly program those many parts and ensure they are and remain in synch and without exploitable bug as the project evolves.

Sky Protocol will allow DApps to seamlessly scale by automatically adapting validator supply to validation demand: The more people want to use Sky Protocol's network, the more the network splits into shards, keeping the overall cost and, at equilibrium, the price, about the same for each megabyte published and kept available for a day. As the network grows, though, the total fee market grows proportionally with the usage of the network, and thus the Sky Protocol token used to split the work and profits of partaking in the network itself grows in value.

Sky Protocol will eventually support bridges to all blockchains. Obviously there are many potential security issues with bridging; each time the bridging functionality is extended, the change will therefore require a community consensus as well as extensive testing (and potentially use of formal methods to qualify the code).

1.3.3. Choosing Cardano

Sky Protocol has the ambition to eventually support all blockchains. Still, Sky Protocol must start with some blockchain, and be careful where to start. The first blockchain should have:

1. A robust consensus algorithm, so Sky Protocol can focus on data availability.
2. Support for smart contract validation, so Sky Protocol can focus on data availability.
3. Issued assets, so Sky Protocol can host its native asset on that blockchain.
4. At the same time, a relative lack of existing scalability solutions on the target blockchain means that there can be a great Synergy with Sky Protocol, especially at the beginning.

These are many reasons why Sky Protocol and Cardano are a great match:

1. Cardano provides a very robust consensus algorithm, based on much academic research, as well as proven in practice by years of deployment in production.
2. Cardano supports smart contracts that are much more powerful than exists on other UTXO-based blockchains, while its data model is based on UTXO rather than Accounts which makes for easier, parallelizable validation of past transactions, which matters for scaling (see our AVOUM whitepaper).
3. Cardano has a builtin notion of issued assets so there is much less aggravation and security risk in using those assets than from e.g. an Ethereum contract.

4. Cardano doesn't yet have a good Layer 2 ecosystem, and even less so a Data Availability Engine, so there is a lot of potential synergy in becoming the first such solution for Cardano.

There is thus a synergy between Sky Protocol and Cardano, both in the short run and the long run: in the short run, Sky Protocol fulfills an immediate market need for Cardano—Scalability via a Data Availability service—while Cardano provides a niche market for Sky Protocol—a welcoming community eager to use the service. In the long run, shared technical choices make for a good partnership: high-level languages, UTXO, powerful contracts, a builtin notion of issued assets, etc.

2. Concepts and Overview

2.1. Decomposing Blockchains

2.1.1. Modular DApps

In our Legicash Whitepaper in 2018, already, we proposed to scale blockchains in a decentralized way, by creating Layer 2 side-chains supported by interactive validation of data posted onto a registry for shared knowledge—techniques now known as "optimistic validation" and "data availability engine", that fill the holes in the previous Plasma Whitepaper. We later named our next (and still current) startup "Mutual Knowledge Systems" in reference to mutual knowledge, the term in epistemic logic and game theory for knowledge shared between all participants in a system—which is what our proposed registry creates.

Since then, systems following this design have been created. Celestia notably implemented this design in a modular way—though their proposed decomposition into execution, settlement, consensus and data availability is slightly problematic, and we propose a better conceptual decomposition below.

2.1.2. Consensus

The best-known service provided by a blockchain (or related architecture) is *consensus*: the ability to have all participants agree in bounded time on what is *the* evolving current state of the system at any moment (or at least, what it was quite recently).

The invention of the Nakamoto Consensus via Proof-of-Work was the innovation by Bitcoin that crucially enabled fully Decentralized Applications. Since then other solutions with different tradeoffs have been discovered, most notably Proof-of-Stake and its many variants.

2.1.3. Validation

We can distinguish from the service of consensus strictly speaking the service of *validation* of the transactions (atomic state changes) that the system executes from one agreed upon state to the next.

The Celestia authors may call this service an "execution" layer, but in general the blockchain need not actually execute anything: for instance, when using zero-knowledge proofs ("zk-proofs"), no transaction execution takes place on the blockchain as such; instead, transaction execution happens outside of the blockchain, privately within clients, that generate proofs; and all the blockchain servers do is check that the proofs are valid, without even having to know what kind of transactions the proofs are about, or how much of what assets they are about. Also, this service is not a "layer" in any meaningful way: there is no way to say that one is above the other or any such thing.

Ways to validate a transaction include (1) indeed having the servers execute every step of every transaction, as in Ethereum, or (2) letting users issue claims, and using an "interactive proof" to establish which of several conflicting claims shall be rewarded or punished, as in the Legicash Whitepaper, and as Optimism and Arbitrum promised to do but never delivered on, or (3) requiring users to publish "non-interactive proofs" using e.g. zk-SNARKs, which can be seen as paying in advance the worst-case price of interactive proofs, in exchange for which reducing the cost and latency of validating transactions.

2.1.4. Data Availability

One last service necessary to complete a Layer 1 blockchains is *data availability*: all participants must see all data so as to be able to validate the data, join the network, see which transactions did or didn't make it, assess the current state of accounts, and generate new transactions.

Layer 1 blockchains implicitly provide and enforce Data Availability, without the issue being usually dwelled upon in whitepapers and documentation: miners who partake in the consensus just refuse to accept, validate and build upon blocks of which they haven't seen the complete contents, and those blocks therefore are never included in the chain. Miners who withhold the contents of their blocks therefore forfeit any fees and reward from those blocks as well as the resources to mine them.

Yet the question of Data Availability becomes an issue to explicitly address when discussing Layer 2 solutions: the Plasma whitepaper (2017) famously mentioned *data withholding attacks* as an issue that could allow failing operators to cause a side-chain to become unavailable, or, depending on the design of the side-chain, even allow dishonest chain operators to abscond with their users' assets.

A solution to this issue is usually called a Data Availability Engine. A popular solution on Ethereum is to publish the data for the Layer 2 on the Layer 1 itself, in a process that Vitalik Buterin calls *rollup*, and that Ethereum now specially supports with its *Proto-Danksharding* (EIP-4844). However, this solution only works if the Layer 1 blockchain is Ethereum (or *mutatis mutandis* if the Layer 1 on top of which a Layer 2 is built supports an affordable similar mechanism); what more it remains somewhat expensive due to the high cost of gas on Ethereum in general (and presumably on any Layer 1 that would become successful enough for a Layer 2 atop it to be useful).

Data Availability is the service that the Sky Protocol will specifically provide through its network of validators.

2.1.5. Bridging

A fourth and last service that a DApp may optionally provide is *bridging* with other systems: communication with these other systems (input and output), actions on these other systems, etc.

Note that this service is *optional*: Bitcoin, and most “Layer 1” blockchains after it, do not usually provide any such service. Transactions processed by the system can only affect or be affected by entities *within* the system, never entities *outside* it. On the other hand, many “Layer 2” solutions do provide bridging:

- Bridges, after which the name “bridging” was chosen, are DApps that connect two or more other Decentralized Systems (blockchains).
- Oracles are DApps that connect Decentralized Systems with Centralized Systems outside the world of blockchains.
- Side-chains are Decentralized Systems with a builtin bridge to a “main chain”, typically a popular Layer 1 blockchain.
- Payment Channels and State Channels can be seen as special cases of Side-chains as above, between a small fixed committee of participants using unanimity as consensus.

The Celestia documentation talks about this fourth service as “settlement”, but that is a bad way to look at it: First, because bridging can be used for much more than settlement, as illustrated by Oracles. Second, because blockchains, whether “Layer 1” or “Layer 2” can settle transactions in their native assets without further service beyond consensus, validation and data availability. Third, because there is no reason why a “Layer 1” blockchain couldn’t directly handle bridging with outside systems as part of its own protocol.

2.1.6. Composing Interchangeable Services

The four services above are largely independent, and it is possible to mix-and-match between several versions of these services as you develop DApps. These services can themselves be parameterized, or layered on top of each other (as in a Side-chain on top of a Main-chain), etc. In the end, rather than a one-size-fits-all blockchain, we can offer a custom kit to build DApps in a modular, composable way.

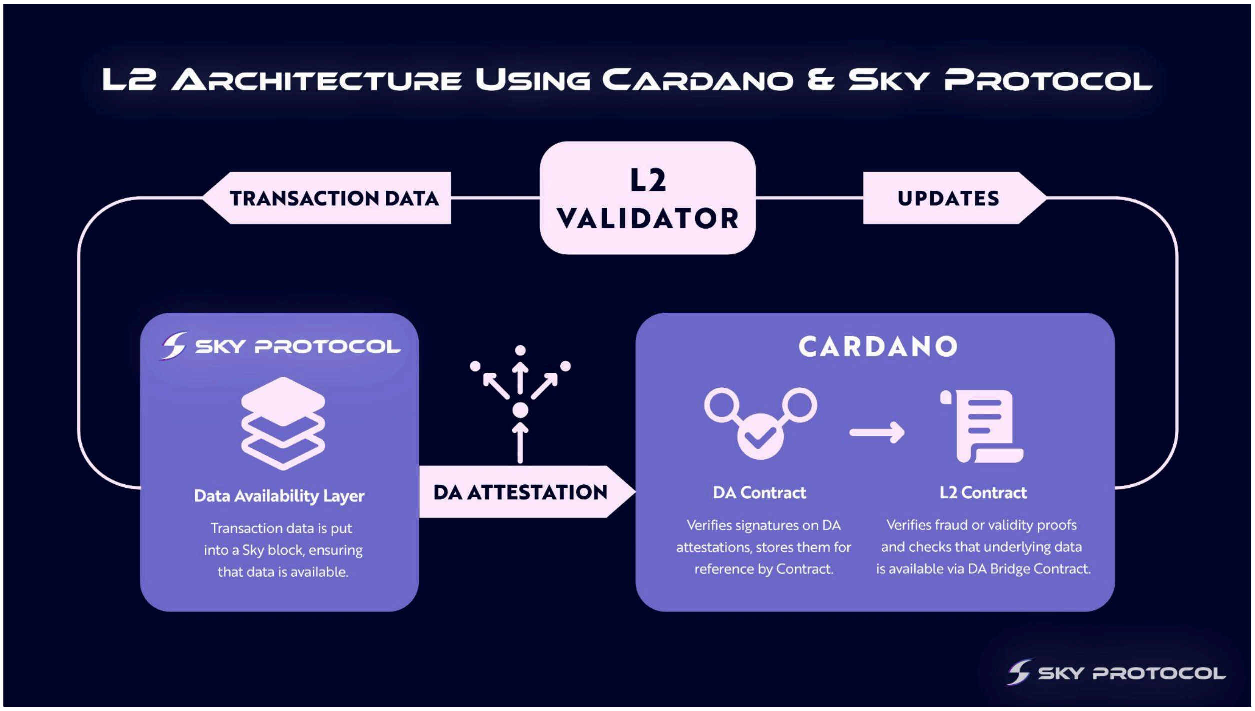
Other projects have tried this before, notably Cosmos and PolkaDot. But they fall short of the ideal in many ways.

First, they use relatively low-level languages that are very well suited for the development of a networked server, but not for the co-development of matching servers and clients and operators and other participants in many roles with matching code on-chain and off-chain, and if there is any bug you lose your money. For this kind of challenge, you will want more of a functional programming language with suitable metaprogramming, so that all the many parts of a DApp can be automatically generated from a single specification. Our team has experience doing this with our language Glow, a language that can generate both smart contract and matching client code from a single specification: we prototyped in Scheme and reimplemented in Agda. See section about *Glow* below.

Second, these ecosystems are largely incompatible with other Decentralized Systems, and are mostly gimmicks to sell more of the Cosmos token ATOM, and the PolkaDot token DOT, respectively. Sky Protocol is designed to eventually work with all existing blockchains rather than requiring users to use Sky Protocol's consensus and associated tokens. Sky Protocol supports and encourages the use of other blockchains and their consensus, either as "the" consensus provider for your DApp, or as chains with which your DApp is bridged. While Sky Protocol will have a consensus of its own to foster the internal security of its own protocol, and that consensus can indeed be used by DApps, it is not the product being advertised, sold and optimized for: the Data Availability service is.

Third, in the particular case of Cosmos, the economic validation is even worse, because each DApp needs its own committee, the capital of which must be raised from scratch as a tiny fraction of the capital already raised by Cosmos. The Cosmos situation was made necessary by Cosmos having failed to define a secure metered VM, and so not being able to securely and efficiently share work between its "app-chains". By contrast, Polkadot having embraced WASM as its secure metered VM can have its consensus validate updates from its "parachains". Sky Protocol will let every DApp developer choose the approach they prefer, but will support and encourage models closer to Polkadot than to Cosmos in this regard.

2.2. Sky Protocol Architecture


L2 Architecture Using Cardano & Sky Protocol
TRANSACTION DATA
L2 VALIDATOR
UPDATES
SKY PROTOCOL
Data Availability Layer
Transaction data is put into a Sky block, ensuring that data is available.
DA ATTESTATION
CARDANO
DA Contract
Verifies signatures on DA attestations, stores them for reference by Contract.
L2 Contract
Verifies fraud or validity proofs and checks that underlying data is available via DA Bridge Contract.
SKY PROTOCOL

2.2.1. Modularity and Options

There will be many ways to build applications with Sky Protocol, though we will start with certain components that provide low hanging fruits for some applications.

2.2.2. Data Availability Engine for Otherwise Existing Layer 2 Solutions

The first component we are building is a pure Data Availability (DA) Engine. This DA can help keep side-chain operators honest and build trust in a decentralized side-chain solution: the side-chain anchoring smart contract can force operators to publish all their data on the DA; then users know they and their agents too can validate the data and build models sufficient to keep issuing side-chain transactions and main-chain transactions to secure the assets. Thanks to their accurate model of the side-chain, they can ensure that they can get their assets out of the side-chain in case of operator failure, whereas no dishonest person will be able to take the assets away.

The model assumes that there is at least one honest watcher, who is somehow incentivized to keep validating the data.

2.3. Attacks and Defenses for Data Availability

The attack model includes bad Data Availability Nodes (DA), Side-chain Operators (SO), Side-chain Watchers (SW), Side-chain Users (SU), who could, sometimes in collusion with each other, try the following attacks, for which we have the following appropriate defenses.

2.3.1. SU Double-spend

Issue: SU tries to double-spend alone.

Solution: Caught by an honest SO.

2.3.2. SO Double-spend

Issue: SO tries to double-spend with assumed SU identity.

Solution: Caught by honest SW if DA is used and working; caught by super-majority honest DA committee if SO tries not to use DA.

2.3.3. SW sleeping on the job

Issue: No one pays SW, or SW is dishonest and does not do its job. There is a coordination problem for SUs who need to collectively ensure that at least one honest SW is running at all times (and resist other attacks such as DDoS, legal attacks, power failures, etc.), though none of them would like to pay the full price of it.

Solution: This issue is usually solved not through technical means as such, but through social norms: side-chain users are expected to semi-randomly subsidize independent reputable SWs, at least only one of which will stay honest. Alternatively, a side-chain may use slower and more expensive non-interactive proofs (i.e. zkSNARKs) instead of interacting proofs, which moves the issue to only one of liveness rather than safety. Eventually, more of the validation can be eventually moved to the DA network (until then it’s not an issue with the DA strictly speaking, though still an issue with modular designs that use DAs).

2.3.4. SO Overload

Issue: SO generates a large volume of transactions until all honest SWs have to give up, at which point they can insert double-spending transactions that no one can verify is bad (if using interactive proofs), and/or that no one else can keep building on (if using non-interactive proofs)

Solution: Have the DA enforce throughput limits for each side-chain and/or “topic”. [*We are not aware of other DA solutions having defenses against this class of attacks.*]

2.3.5. DA complicit in Data Withholding or Overload

Issue: Dishonest DA nodes help dishonest SO in trying to prevent SW from doing their job.

Solution: Have other DAs watch each other, with slashing of stake from dishonest ones. This requires the network to produce or at least use actual consensus, not just Data Availability (though Data Availability can come first, and turn into consensus later; or the two can remain separate). [*Celestia provides client-side "data availability sampling" checks to detect such conditions, but these checks come too late to save any of the user's assets, provide no means to sanction the dishonest DA, and in cases of shard data unavailability cannot distinguish between bad DA and bad SO.*]

2.3.6. DA collecting fees without working

Issue: Dishonest or otherwise broken DA does just enough work not to get caught and collect fees, but does not otherwise actually serve data to users, thus not contributing to security, yet without actively trying to steal funds.

Solution: Make the future DAs that watch the current DAs as indistinguishable from regular users as possible. We can use measures such as:

- Have the watchers be not other members of the current committee, but yet-unrevealed (yet self-identified) members of a future committee.
- Minimally cooperating with other DAs to avoid punishment while denying data to actual users, or otherwise being malicious in modifying their behavior based on who is talking to them—avoided by making DAs indistinguishable from regular users by (1) having unrevealed DAs from a future period watch DAs of the current period, (2) designing the query protocol and the access patterns so it's hard to distinguish a DA from a user, e.g. by having queries go backwards as well as forward or having fixed granularity, (3) optionally, but probably importantly if a client makes more than one request, using Onion routing to make requests indistinguishable by address of origin (at the cost of extra latency and reduced throughput).

2.3.7. DA who just relay queries

Issue: a lazy DA may "just" relay queries to other committee members without doing any work of its own, counting on 67% of the other DAs to do the job, and collecting fees without doing any real work, without actually contributing to network security.

Solution: Make it more costly to relay queries than doing the work, by having each DA carry a different fragment of the data (using some error-correction code), such that a DA would have to query two-thirds of the other DAs to reconstitute its fragment.

2.3.8. DA reneging on what it claims to publish

Issue: DA telling SO his data was accepted but not actually including it.

Solution: Require a commitment from DA, and SO publishing a fraud proof if there is a lie, and other DAs and/or contract can punish the DA.

2.3.9. DA refusing service to SO

Issue: DA for some reason legitimate or il- will not serve SO.

Solution: Allowing SO to get away with only 67% operators having the data—though that will cost *more* (see below).

2.3.10. SO trying to skimp by not contacting all DAs

Issue: Since only 67% of DAs are required for the data to be considered available, a cheap SO might be tempted to only contacting 67% of DAs and save on 33% of fees.

Solution: SOs pay the network using SKY token, not individual DAs, and may exchange USD (or whatever) for SKY at the last minute. Network accounting of payment is done in SKY. The payment doesn't directly go to the DA, but instead to a common pool. After Sky Protocol reaches consensus, each DA who signed the transaction on time receives a fee proportional to the number of DA who signed, and the SO further receives a proportional rebate in SKY, while the remaining is burned (proportional to those who didn't sign on time).

2.4. Open Question

2.4.1. Who talks to all the DAs?

Is it the SO's responsibility to contact each DA, or the first DA's responsibility to transitively transmit the data to each other DA? The former makes slightly more sense in a pure DA service, while the second makes slightly more sense if optimizing for consensus.

If the SO wants a prompt answer, the former, which also relieves oh so slightly the inter-DA communication pipes. Should the SO pay less if he does that work? Pay more to get the prompt proofs of inclusions?

Register transactions on a consensus that happens after the data is merely available, and pay committee members and reimburse fees based on how many DAs signed the data on time. If signatures are missing, the money is burned (or *maybe* sent to a charity or common pool), not reimbursed to the user.

Solved by increasing cost (decreasing post-consensus rebate) if fewer than 100% of DAs were contacted.

- SO refusing (or incapable?) to use enough DA operators to guarantee availability
- DA trying to relay queries to other members to supply actual availability (same encoding trick as filecoin?)

In response to this challenge, Sky Protocol is committed to constructing a tailored Data Availability Solution aligned with the requirements of Cardano. Our approach involves publishing data in a format conducive for Cardano smart contracts to seamlessly validate and integrate into Layer 1 contracts. Rooted in Cardano and underpinned by its own Cardano-based token, our network reduces reliance on external networks and minimizes trust assumptions. Our network enables multiple Cardano-based DApps to leverage the same Data Availability network, thus pooling resources and capital for network validation. Additionally, our network will progressively introduce censorship-resistant features, including time-locked blinding of data to prevent selective censorship by committee members, and RAID5-style coding of data to ensure data confidentiality. Moreover, our network will support DApp validation through both "optimistic" interactive games and "pessimistic" zero-knowledge proofs.

This comprehensive approach empowers DApp developers to craft high-speed, low-latency applications or highly censorship-resistant, low-latency applications, all with elevated throughput and reduced costs, anchored on the Cardano blockchain.

2.5 Mutual Knowledge vs Common Knowledge

(as illustrated in this episode of Friends).

We take a unique approach in building this protocol. Typically, blockchain consensus establishes an official record of historical events occurring on the network and their impact on the ledger. Every user in the network unanimously agrees upon this record, and there exists a mutual understanding that all parties are aware of it. This shared understanding, termed as Common Knowledge, has been pivotal in the success of cryptocurrencies and blockchain-based systems. However, scalability remains a significant challenge inhibiting market growth, particularly evident in Cardano's prioritization of security and decentralization over immediate scalability.

Conversely, Mutual Knowledge occurs when every participant possesses the same historical knowledge but lacks awareness of each other's awareness. In numerous cases within the blockchain space, Mutual Knowledge could suffice to ensure secure transactions and could significantly expedite processes. Mutual Knowledge represents the next evolutionary step towards constructing a decentralized internet, albeit necessitating technical solutions, as outlined in this document.

The Sky Protocol scaling solution aims to address scalability challenges by imbuing every facet of its ecosystem with Mutual Knowledge, thereby enabling the scaling of any DApp on any blockchain. Sky Protocol is a modular data availability network that securely scales with the number of users, making it easy for anyone to launch their own blockchain.

In this model, Sky Protocol is only responsible for ordering transactions and guaranteeing their data availability, similar to reducing consensus to atomic broadcast. Data availability sampling provides an efficient solution to the data availability problem by requiring resource-limited light nodes to sample a small number of random shares from each block to verify data availability.

Interestingly, as more light nodes participate in sampling, the network can safely handle a greater amount of data. This means the block size can increase without equally increasing the cost to verify the chain, enabling greater scalability and efficiency.

And last, Sky protocol posts state updates to the main chain, but doesn't overwhelm it with all the data (mutual knowledge : people know what is happening on the chain but don't need a systematic proof that everyone knows that they know). For this reason Sky Protocol is below a side chain level but slightly above a layer 1, because it has its own consensus.

2.6 Sharding and topics

In the beginning, it will not be necessary to use sharding for Sky Protocol to be optional (or we could start with only one shard) if we want to reduce the complexity of our work.

In the long run, what we could want is to have each shard being a rollup for a different Layer 1. If and only if this happens, the Sky Protocol network will be divided into multiple independent parallel shards, each with its own structure and security tradeoffs. Each shard can delegate consensus management to its own layer 1 network. Applications choose the shard they need to run on, and shards follow broad validation rules.

Similar to some other blockchains and pub/sub systems, Sky Protocol will extensively use Topics to maintain a clean distributed system, handle data processing, and manage message brokering. A "topic" is a logical category or channel where producers publish messages and subscribers consume them. This separation allows decoupling of message producers from consumers, enhancing both privacy and scalability. This aligns with our Mutual Knowledge concept, where producers and consumers do not need to know each other but only need to agree on the topic name. For example, multiple consumers can read from the same topic, distributing the load of message processing. This approach simplifies data management and processing within Sky Protocol.

Each topic will be further divided into shards, enabling versatile use of a topic's data across multiple partitions to improve scalability and performance. Each shard can reside on a specific side chain with its own validation rules, allowing parallel processing of published data, which enhances throughput and fault tolerance.

To further enhance scalability, we plan to implement a cross-shard interaction system in the future. We have been familiarized with the safety challenges of such a technology when we built one for Harmony One (Whitepaper here, and Gitlab here)

These technical choices will benefit all applications using smart contracts, DApps, and event listeners by facilitating their operations and improving overall network efficiency, but many of these choices will be given to the users so they can develop applications using the right compromise between performance and safety in their own business logic.

In the same order of ideas, when considering data storage strategies, there are two primary approaches: keeping large unvalidated data in shards with few copies, or keeping validated data in many copies. Storing unvalidated data in shards with few copies can significantly enhance data availability (DA) and provide natural resistance to censorship. However, this approach does not contribute to consensus-level censorship resistance unless data availability is integrated into the consensus mechanism itself. On the other hand, keeping validated data in many copies bolsters consensus-level censorship resistance but may not be as efficient in scaling data availability. Each approach has its trade-offs between scalability, data availability, and censorship resistance, depending on whether data validation and integration with the consensus process are prioritized.

At last, we will implement a safety mechanism to prevent users from running topics of different security levels on the same machine.

3. DApp Component Marketplace

In its final form, Sky Protocol will have a marketplace wherein DApp developers can create DApps that can take advantage of the strengths of different blockchains for different parts of their DApps. For example, building a DApp that provides anonymity to its users through Monero, but with the security of Cardano and the scalability of Algorand.

This setup will allow DApps running on any chain to scale independently of the blockchain they use for finality.

For this to happen, the Sky Protocol ecosystem is based on the following parts:

1. A Mutual Knowledge Base
2. Generalized State Channels
3. Side-Chain markets
4. Account View on UTXO models
5. The Glow programming language

3.1 Mutual Knowledge Base (MKB)

The Mutual Knowledge Base (MKB) serves as a universal Data Availability Engine capable of supporting an array of Side Chains and State channels across multiple blockchains, with an initial focus on Cardano. Essentially, the MKB functions as a distributed registry processing transaction outcomes among users without prior history knowledge. It then hashes these outcomes and transmits them back to the current blockchain. Functioning as a parallel network of registrars, the MKB constructs Mutual Knowledge—a publicly accessible record of information accessible to all. Compared to Common Knowledge, Mutual Knowledge offers swifter and more cost-effective scalability, presenting a superior solution for network expansion. Furthermore, the MKB's scalability surpasses that of a consensus blockchain, with individual DApps receiving their own rate-limited network shard. Although the scaling solution still necessitates Consensus (Common Knowledge) at specific junctures, it significantly conserves resources and computation time. Moreover, the MKB serves as a pivotal intermediary between Side Chain Markets, enabling the exchange of transaction data while preventing block-withholding attacks, crucial for Plasma-like side-chain designs. The Sky Protocol Scaling Solution leverages the MKB's capabilities to bootstrap its operations, utilizing a simplified variant where both services are provided by the same entities. Powered by its utility token (SKY), the Sky Protocol MKB maintains its Side Chain Market, accelerating payment to registrars. With Proof-of-Stake (PoS) consensus, the MKB ensures network security, with additional protections such as asset-indexing and continuous defragmentation enhancing overall robustness. An example implementation of the Sky Protocol design showcases its ability to facilitate fast transactions via Side Chains while bolstering security through the MKB. While still in development, this implementation underscores the potential of the Sky Protocol to process high-volume transactions efficiently and securely.

3.2 Generalized State Channels

State channels are a layer-two blockchain scaling solution. They allow improved throughput and latency for the main blockchain, by allowing uncontested transactions between multiple parties to settle off-chain, and a quick resolution when a transaction is contested because of the body of transactional cryptographic evidence that they accrue, so that in either event, in-progress contracts can continue under the consensus provided by the main chain.

3.3 Side-chain Market

Another approach to bolstering the scalability of a blockchain is through side chains. These chains allow a group of participants to execute a series of smaller transactions off the main blockchain, only committing the initial and final states back to the main chain. Our scaling marketplace endeavors to establish a network of state channel and side chain operators, each overseeing a side chain in a predominantly centralized manner, thus facilitating rapid transaction recording for specific DApps. Unlike traditional centralized applications, however, each side chain is publicly disclosed as a verifiable data structure, enabling operators to audit one another and hold each other accountable for any instances of failure or abuse.

To ensure mutual accountability, operators within a Side Chain Market adopt the Consensus-as-Court paradigm pioneered by TrueBit and Plasma. Assets are managed on the “main chain,” which provides consensus (e.g., Cardano), through a “Smart Contract.” This consensus is solely utilized for settlement and dispute resolution in case of double spending or contract breaches, with the Smart Contract code effectively serving as the “smart judge.” Participants (operators, registrars, and users) can lodge claims against the Smart Contract, which are inexpensive when substantiated but costly when unsubstantiated. Following a challenge period (and potential challenges), the claims are validated (or invalidated), and if validated, users can retrieve corresponding assets from the Smart Contract based on those claims. In the event of a dispute, this process unfolds automatically, resulting in transgressors being penalized and victims compensated.

Users retain the option to exercise their “right to exit,” unilaterally repudiating any unsatisfactory operator and switching to another service provider without incurring switching costs, queries, conditions, or concealed fees. Operators failing to meet their contractual obligations automatically forfeit all their users in favor of remaining operators, in addition to losing their bond. Malicious operators are unable to steal or freeze their users’ assets but can merely stall them until they lose their users.

A Side Chain Market can be tailored for each specific Distributed Application (DApp). Despite incurring a marginal additional cost per transaction, with the sharing of capital costs, any number of DApps can operate on the same general-purpose Side Chain Market.

The Side-Chain Market introduces the security assumption that at least one operator is honest. Furthermore, the security of a Side Chain Market necessitates a Data Availability Engine to prevent any “block withholding attack” on the Smart Contract controlling the deposited assets. While any existing consensus blockchain could theoretically serve as such an engine (as in the Ethereum “rollup” design), Sky Protocol opts for a dedicated Mutual Knowledge Base for superior scaling at a reduced cost.

This solution holds significant benefits for Cardano, as it enables rapid transaction processing and enhanced scalability for DApps operating on the platform, aligning with Cardano’s vision of providing a secure and efficient blockchain ecosystem. Additionally, the accountability mechanisms inherent in the Side Chain Market contribute to the overall security and trustworthiness of the Cardano network, fostering greater confidence among users and developers alike.

3.4 Glow Language

Glow is a programming language tailored for decentralized applications (DApps), prioritizing safety, user-friendliness, and portability across various blockchains. For a Cardano Layer 2 protocol focused on data availability, integrating Glow offers several key benefits. Firstly, Glow's emphasis on safety ensures that newly implemented functions undergo thorough audits, minimizing the risk of vulnerabilities in interactions between users. This aligns with the protocol's objective of providing a secure platform for storing and accessing data. Secondly, Glow's user-friendly design simplifies the development process, making it easier for developers to create DApps that meet user expectations. This accessibility can attract a broader audience to the protocol, enhancing its adoption and usability. Lastly, Glow's portability enables DApps written in the language to run on any blockchain, allowing the Cardano Layer 2 protocol to leverage resources from multiple blockchains simultaneously. This flexibility ensures that the protocol can adapt to the evolving blockchain landscape and reach users across different platforms, further enhancing its utility and scalability. Overall, integrating Glow into a Cardano Layer 2 protocol focused on data availability streamlines development, enhances security, and increases accessibility, ultimately contributing to the protocol's success and growth.

3.5 Account view on eUTxO.

AVOUM (Account View On UTXO Model) is a technology that combines the strengths of the UTXO (Unspent Transaction Output) and Account models in blockchain architecture. For Sky Protocol, a Cardano Layer 2 focused on data availability, AVOUM offers several benefits.

Firstly, AVOUM enhances the security of the blockchain by leveraging the efficiency of UTXOs to check past transactions in parallel, enabling faster addition of new full nodes and thereby strengthening the network's security. Additionally, AVOUM introduces the capability for users to queue multiple future transactions concurrently on the same smart contract, facilitating a more dynamic and robust contract ecosystem. This feature is particularly advantageous for Sky Protocol, as it enables the development of richer and more versatile smart contracts, enhancing the platform's functionality and utility for users. Moreover, by combining the UTXO model for past transactions with an Account view for future transactions, AVOUM enables the implementation of public smart contracts without sacrificing decentralization. This aligns with Sky Protocol's goal of providing a secure and decentralized platform for storing and accessing data. Overall, integrating AVOUM into Sky Protocol enhances its scalability, security, and functionality, positioning it as a competitive and innovative solution in the blockchain space.

4. Governance

Sky Protocol's governance model operates through a decentralized autonomous organization (DAO) structure, empowering participants to influence the platform's development and decision-making processes. Participants can vote on various proposals and initiatives through the Sky Protocol (SKY) governance mechanism. These proposals encompass a wide array of topics, including protocol upgrades, fee adjustments, parameter changes, and resource allocation.

However, there are parts of the roadmap that are not dependent on the users’ preferences and that can’t afford to be built in a bottom-up way. Thus these parts of our roadmap will not be submitted to any kind of vote. Participants can cast their votes on these proposals using SKY tokens, with each token representing a proportional voting power.

The governance process is designed to facilitate community consensus and ensure that the platform's evolution aligns with the collective interests of its stakeholders. The voting process will be open source and on-chain for complete transparency and verifiability of all governance votes.

5. Development Roadmap

Milestone	Timeline
Ecosystem Partners	**Q2 2024**
Cardano L2 Testnet	**Q3 2024**
Audit	**Q3 2024**
Cardano L2 Mainnet	**Q4 2024**
Cardano L2 Expanded Features	**Q1 2025**
Sky Protocol Expansion	**Q2 2025**

5.1 Testnet deliverables

For deploying our L2 testnet, we will use many components from our already built Glow Language. This is publicly available on Github.The next milestones are the following.

Deliverable #1: Contract that interacts with a Centralized Data Availability Operator. It will be a simple Cardano contract that depends on specific data having been published on a *centralized* Data Availability service. For instance, we will write a contract that pays a certain amount if a preimage to an agreed-upon hash was published on the Data Availability service. This milestone will thus demonstrate how a simple Cardano DApp can make use of a Data Availability service.

Deliverable #2: Contract that interacts with a Decentralized Data Availability Committee. The third milestone will be a contract that depends on specific data having been published on a decentralized Data Availability service. We will modify the DApp above to work with a *decentralized* service. The DApp will be largely unmodified, but the data availability validation library it uses will be more sophisticated.

5.2 Mainnet deliverables

Deliverable #3: End-to-End integration of contract using the Data Availability Network.

This can be considered as the Sky Protocol MVP. What we want is the data availability engine and the ability to publish data. This will already allow all DApp developers to have a functional scaling solution. We might only have one topic available.

5.2.1 Stretch goals for SKY token sale

- Multiple topics
- Switching to proof-of-stake instead of proof-of-authority when Sky Protocol has reached a significant amount of users
- More versatile validation rules for specific side-chains/shards and topics: update of acceptance script, choosing between stateful or stateless interactions, configuring the remanence (duration that data remains available)

5.3 Expansion - phase 1

- Portability to other chains with a native bridge to the existing Sky Protocol.

5.4 Expansion - phase 2

- Development of a flourishing Dapp ecosystem using Sky Protocol’s scalability with geographic adjustments for the countries that need to work with regulatory constraints.
- Implementation of our bug-bounty and grant system.

6. Tokenomics Introduction

SKY is the utility token for the SKY protocol. A detailed explanation of SKY tokenomics will be released as a separate document and the link will appear in this section.

7. Team

Our team is comprised of seasoned blockchain and Cardano developers, with a track record of innovation and leadership in the industry. Notably, our team spearheaded the development of the Glow Language, previously commissioned by IOHK, and pioneered the Account View on UTXO, or AVOUM, technology.

Recognized for our contributions, we were honored as Catalyst grant recipients for our work on Formal Verification for Glow and our comprehensive study of AVOUM's potential impact on Cardano. Moreover, our team has shared insights and expertise at various Cardano Conferences. Charles Hoskinson commented on our AVOUM model and his thoughts on the technology in this video.

Beyond the realm of Cardano, our team has made contributions to the academic discourse, publishing papers on formal verification and blockchain technology. Those publications can be found here: https://mukn.com/publications-from-our-team/

Our team has also earned grants at several Catalyst funds, and has developed other pieces of technology for Harmony One, Filecoin, XRP / Ripple, Nerv0s, Sequentia and has a live crypto-currency payment app on Salesforce.

In addition to its usual team, Sky Protocol has also a partnership with Blink Labs, who are known for their extensive knowledge of Cardano and their proven track record as developers in this ecosystem.

Our superpower is fluency in many domains ranging from theoretical computer science, programming language design and implementation, logic, reflection, distributed systems, network protocols, cryptography, game theory, economic mechanism design, cryptoeconomics, finance, law, adversarial thinking, and more. Our secret weapons are the programming languages Gerbil Scheme and Cubical Agda, enabling us to achieve what others cannot. Our team will grow in numbers after the token sale of Sky Protocol on Cardano, and this document will be updated.

Faré / François-René Rideau - Lead Architect

Faré is the President and Chief Scientist of MuKn, a consultancy creating innovative decentralized solutions for both public and private blockchains.

President and Chief Scientist of MuKn, François-René “Faré” Rideau has more than 25 years of experience building programming languages and distributed systems. He notably proved the correctness of a (centralized) payment protocol early in his career.

Alumnus of École Normale Supérieure, Université de Nice, and Télécom Paris, Faré went to the United States and worked as a Senior Developer for companies such as ITA, Google and Bridgewater Associates.

While working in the industry, he notably maintained and rewrote ASDF, the build system at the heart of the Common Lisp open source community; he also kept publishing academic papers and speaking at programming language conferences.

Eventually, his interests in economics and software security converged with his experience in open source software and formal methods and he started working on Layer 2 solutions for the blockchain. Faré was also one of the original developers of the Alacrity language that later was forked to become Reach on Algorand.

Other technologies designed by Faré but not previously referred to in this whitepaper include:

- **UVOAM:** Enables private and composable UTXO-style transactions on account blockchains.
- **Sigurity:** Provides programmable multisigs indistinguishable from regular signatures to outsiders, yet fully authorized, audited, and logged within the corporation.
- **IWillPersist:** Ensures that DApp clients' data persists securely and privately on redundant blinded remote servers.
- **Durabo:** Empowers users to control their own uncensorable decentralized message feeds.
- **Decenstake:** A censorship-resistant leaderless Proof-of-Stake consensus algorithm.
- **TrieZip:** Accelerates the Merkleization of large data structures by 10x.
- **EVMBatch:** Allows atomic batching of transactions in a single group transaction.
- **MetaCreate2:** Ensures that contracts can have the same address on all EVM blockchains.
- Efficient implementations of Yield Farming, Automated Market Making, Lotteries, or ERC20 contracts.
- **Legicash:** (Designed at a previous startup) Automates the generation of "proof-of-fraud" machinery for optimistic side-chains.

Gauthier Lamothe - Head of Operations

Gauthier specializes in business development and communications. He has been a head of operations and CEO of a few successful companies in France and has worked as a team leader and business developer in previous blockchain projects for MuKn and the Free Republic of Liberland.

Gauthier is an expert in all areas of media, and is passionate about using blockchain to solve global injustices. Former film producer, Gauthier produces MuKn's "Mutual Knowledge" Podcast and educational information on blockchain technology and business, interviewing the world's leading experts on using blockchain technology for social change.

Areas of Gauthier's previous works include decentralized justice systems, tokenization of governance, and use of cryptocurrency for decentralized systems of freedom.

Doubling his career with psychotherapy and coaching, Gauthier acts as a trainer and advisor for many companies (in France or in other countries) in fields such as biotech, insurance, wellbeing, banking, nuclear physics, fast-food, and goods distribution.

Alex Hochberger - Tech Advisor

Alex M. "The Hoch" Hochberger has over 20 years of experience integrating technology and business systems to drive real growth. He notably spent an extensive amount of time building and advising startups from formation to exit. His first startup was formed with his MIT roommate (sold 7 years later) through doing technology turnarounds of portfolio companies at Z9 Ventures, a Miami-based venture group.

One of the original MiamiTech "OGs," Hoch was a mentor with Incubate Miami, the original Miami Accelerator. Now he is an advisor with Cryptan Labs Web3 and Blockchain Accelerator in Miami.

Alex's career has included building out and configuring Enterprise and Startup versions of CRM, ERP, and Data-storage systems, from the Web1 Dot-com Bust through the Web3 era, with expertise using technology to solve real world problems. His stints have included CTO and CMO roles in consumer finance, health care, and consumer product space.

Last but not least, he is the CEO of Web3 Enabler, offering crypto-payment solutions on Salesforce

Zoe Braiterman - Data Safety Expert

Zoe Braiterman has experience ranging from cyber security, data science and system architecture to product development.

She is strong with an extensive experience as a security consultant and research associate for PurePoint International, Analyst and Lead Data Architect for multiple companies and an active prominent member of the OWASP Foundation.

Passionate about helping startups to scale, she also has experience as a teacher and business manager. She now secures the whole MuKn information ecosystem.

Donald Fisk - Developer

Donald has a BSc in Physics and Astronomy from Glasgow University,and an MSc in Telecoms Engineering from Kings College, London.

Donald began programming in 1981, and became interested in Artificial Intelligence and Lisp shortly afterwards. Lisp is his favorite language, but he has also programmed in C, Java, Python, Prolog, Pascal, various assembly languages, and more recently in Solidity.

He has developed AI rule-based systems for fault recognition and routing in telecommunications systems, musical score generation from MIDI input, phonetics, and web page layout, and also used Lisp for document processing, and to develop a visual dataflow programming language, Full Metal Jacket.

He has had papers published on MORSE, a collaborative filtering system for movie recommendation, and on Full Metal Jacket. He has worked mostly in research and development, and is inventor/co-inventor on six patents in telecommunications, computer graphics, and musical score generation.

He has a personal web page at http://fmjlang.co.uk/, which has links to his papers and patents.

His blockchain experience started in 2021 when he worked on a few development projects for MuKn, including a data availability engine.

Marcin Grzybowsky - Developer

Marcin has been working professionally in software development for more than 15 years.

After using Agda and Idris for personal projects, he became captivated by dependently typed languages and Homotopy Type Theory and has been using these technologies for the benefit of commercial projects Since 2018.

At MuKn Marcin is notably tasked with the implementation of the Formal Verification functionality of Glow, and the development of machine-checked proof of its properties, but his work also includes the development of other technologies in our own R&D lab.

Multiple Laureate of the Polish Physics Olympiad (2004,2005), Marcin has experience in many different fields ranging from Artificial Intelligence, Domain Specific Languages, and has also cofounded startups.

Marcin is also experienced in the field of AI, involved in the FormalFoundry project, that uses formal methods to prove the safety of AI applications and AI to improve the usability of formal methods, offering technologies that combine machine learning and software verification to the wider public.

Alexander Smart - Chief Legal Officer

CEO of MuKn, Alexander Smart has a B.A. from UChicago with an emphasis in Political Science and a minor in Economics, and a J.D. from Pepperdine Caruso School of Law. He is licensed to practice law in California, New York, and Massachusetts.

Alexander has always thought fast, but learned to think deep and sharp at UChicago. After studying law at Pepperdine, he spent nearly fifteen years guiding executives and decision makers through litigation, in matters ranging from shoplifting and speeding tickets to multi-forum international investment bank disputes.

His practice honed his ability to quickly assimilate and master new information, and deliver that information clearly at any level of sophistication. Tiring of courthouses, he found his skills were readily applicable and desperately needed in the blockchain space.

Informed by two years of volunteer work he did in Northeastern Brazil in his college years, he joined MuKn to help use Web3 to give people in every part of the world access to modern financial services.

Peter Hubshman - Chief Financial Officer

Peter is a finance and operations expert focusing on early round startups. With origins in private equity, fund management and leveraged buyouts, in the early 2000's he operated Internet Real Estate Group, a Web 2.0 studio in Boston which successfully developed businesses including Creditcards.com; Phone.com; Luggage.com; Jeans.com and a dozen other early primary domain businesses. There, his pioneering team of engineers created some of the earliest successful affiliate marketing and advertising platforms on the Internet, and were early experts in search engine optimization.

In 2009 Peter became CFO of Digiport Data Centers in Miami FL until its sale in 2013. There he developed business plans and economic models for all of Digiport's spinout startups, including:The collaborative consumption platform, Boatsetter.com (Air BnB for yachts, captains included); SaaS startup, Itopia.com (early cloud-ware for small-midsized professional offices and now, .edu); and artworld upstart, Blackdove.com (a digital arts platform for digital artists, restaurants, residential, and corporate). In 2018 Peter was Management Consultant to, and Interim CFO at TheDroneRacingLeague.com (e-sports, media, and drone technology). There he helped prepare the management team and crew for a successful C round led by a prestigious investment bank in 2019. Later in 2019, he became Interim CFO of Gemic.com (Brand Strategy for Fortune 100's). There he led the financial team supporting their successful A round with Bocap, a private equity firm in Finland.

Peter joined MuKn in Q1 2022 as CFO, Co- Founder and a Board Member. He remains a Member of the Board, and is currently Finance Advisor while on assignment as CXO to a new MuKn studio spinout, FormalFoundry.ai (Dedicated to ensuring the correctness of AI models at scale). Peter studied economics at Tufts University and has a Masters in Public and Private Management from the Yale University School of Management.

8. Security

Sky Protocol embodies an open-source approach, elevating security standards in line with the principles of cryptocurrency. This commitment to transparency and scrutiny mirrors the ethos of decentralization and trust inherent in cryptocurrency networks. With openly accessible code, a diverse community of developers, auditors, and users can meticulously examine it for vulnerabilities, errors, or malicious elements, fostering a culture of collective responsibility and collaboration. Conversely, closed-source models restrict code visibility, limiting the breadth of perspectives available for security evaluation and potentially compromising the integrity of the protocol. By embracing open-source principles, Sky Protocol promotes a secure and resilient ecosystem, aligning with the core tenets of cryptocurrency networks.

Regarding audits, as an integral part of the development process, the protocol engages external auditors to ensure robust security measures are upheld. These auditors conduct independent assessments, identifying potential flaws and conducting thorough code reviews to safeguard the protocol's security and reliability. This commitment to rigorous auditing reflects the protocol's dedication to maintaining the highest standards of security and transparency within the cryptocurrency space.

9. Appendix: Bibliography

B.1 Previous Relevant Publications by Our Team

François-René Rideau, “Chenilles Whitepaper”, whitepaper, 2023.

François-René Rideau, “AVOUM: Account-View-on-UTXO-Model”, whitepaper, 2022.

François-René Rideau, “Durabo: Unstoppable Message Feeds, 2021”, whitepaper, 2021.

François-René Rideau, “Simple Formally Verified DApps—and not just Smart Contracts”, EthCC[3], 2020.

François-René Rideau, “Glow Whitepaper”, 2020.

Jay McCarthy and François-René Rideau, “Alacrity: A DSL for Simple, Formally-Verified DApps”, DevCon5, 2019.

François-René Rideau, What do Formal Methods actually Guarantee?, 2018

François-René Rideau, Language Abstraction for [V]erifiable Blockchain Distributed Applications, IOHK Summit, 2019.

François-René Rideau, “Composing Contracts without Special Provisions — using Blockchain History” Hackernoon, 2019.

François-René Rideau, “Binding Blockchains Together With Accountability Through Computability Logic”, LambdaConf 2018.

François-René Rideau, “Why Developing on Blockchain is Hard? - Part 2: Computing Proper Collateral”, Hackernoon, 2018.

François-René Rideau, “Why Developing on Blockchain is Hard? - Part 1: Posting Transactions”, Hackernoon, 2018.

François-René Rideau, “Legicash: Binding Blockchains Together through Smart Law”, Legicash Whitepaper, 2018.

François-René Rideau, “Climbing Up the Semantic Tower — at Runtime”, Off the Beaten Track Workshop at POPL, 2018.

B.2 Other Relevant Publications

“Flashbots Transparency Report”, February 2021.

Dan Robinson and Georgios Konstantopoulos, “Ethereum is a Dark Forest”, 2020.

“Flash Boys 2.0: Frontrunning, Transaction Reordering, and Consensus Instability in Decentralized Exchanges”, 2019.

Mustafa Al-Bassam, “LazyLedger: A Distributed Data Availability Ledger With Client-Side Smart Contracts” (Celestia Whitepaper), 2019.

Joachim Zahnentferner, “Chimeric Ledgers: Translating and Unifying UTXO-based and Account-based Cryptocurrencies”, March 2018.

“How the winner got Fomo3D prize — A Detailed Explanation”, SECBIT Labs, August 2018.

Joseph Poon and Vitalik Buterin, “Plasma: Scalable Autonomous Smart Contracts”, 2017

Satoshi Nakamoto, “Bitcoin: A Peer-to-Peer Electronic Cash System”, 2009.

https://x.com/inputoutputhk/status/1366838241580703744

https://iohk.io/en/blog/posts/2021/10/29/mithril-a-stronger-and-lighter-blockchain-for-better-efficiency/

This page is intentionally left blank.