FireDex

fidx.xyz

August 12, 2024

Contents

1 Introduction

Cryptocurrency promises to reimagine how finance and payments work. The technologies that survive will be those that find a way to serve customers better than the status quo.

One major theme is to minimize trust in centralized actors. The world of traditional finance follows the premise of "trust me, because even though I *could* do bad things, I usually don't". Cryptocurrency projects counter "you don't have to trust me, because open source code and engineering prevent anyone from cheating".

Decentralized Finance (DeFi) is the future of reliable financial services.

Today many large DeFi projects offer traditional financial services like trading and loans, via blockchain smart contracts, notably in Ethereum and related ecosystems. However, this space is plagued by an insidious and fundamental problem called "Miner Extractable Value" (MEV) [14].

On Ethereum all trades are submitted transparently to the blockchain nodes before they are final. The blockchain nodes can see them before they happen, and can interfere with, reorder, or front-run the user's trades. Today the majority of Ethereum nodes use

Flashbots software to automate this process [28]. The profit from this activity is called MEV. This is a noticeably worse state of affairs than even in traditional finance.

For instance, imagine competing against the New York Stock Exchange itself when placing stock market orders; the ordinary trader would lose every time. This is what it's like for most users trading on DeFi systems right now.

There has to be a better way.

Enter MobileCoin

The MobileCoin network has ideal properties to fix this problem. MobileCoin's network is unique in that every node contains an SGX enclave[1], and crucially, every transaction is encrypted for the enclave when it is submitted. Consensus happens on opaque hashes of transactions. None of the results are revealed to the node operator until they are final.

In MobileCoin this design was chosen to protect user privacy. It has the side-effect of preventing MEV, since the "miners" cannot see the transactions before they are final.

MobileCoin transactions settle in less than a second and can be built quickly on low-end mobile devices. This is a network aimed at payments that currently doesn't have a good on-chain method for customers to convert one currency to another, a requirement for global payment processing.

To take advantage of the network's properties for currency conversions and for DeFi, we proposed a new feature called "Signed Contingent Inputs". This feature was implemented in MobileCoin Improvement Proposals #31 and #42.

To offer a trade to someone, a MobileCoin network token is signed for so that any other person can use it as input in a transaction, as long as they produce a transaction output for the signer of the desired value and token type. This system supports partial-fills, which is unique for peer-to-peer on-chain token swaps.

MobileCoin meets FireDex

Practically speaking, these are the elements required to build a world-class trading experience: fast settlement, almost zero transaction cost, self-custody, and end-to-end encryption.

FireDex is a layer-two network[2] on top of MobileCoin that helps users broadcast offers and find offers of interest. Market-makers and traders compete to broadcast the best offers, so that customers who want to buy a cup of tea priced in Nigerian Naira, but who only have United States Dollars in their MobileCoin wallet, can use these offers to swap

[1] SGX is a hardware security technology developed by Intel. An enclave is an environment in which code can execute at native speeds, while enjoying memory protections guaranteed by the Intel CPU. Particularly, working memory is encrypted and decrypted transparently as it enters and leaves the CPU. For a deep-dive, see "Mechanics of MobileCoin" [21] references therein.

[2] "Layer-two" is blockchain jargon meaning a decentralized network that builds on top of a "layer-one" network and adds capability and value above and beyond the layer-one network. This terminology was popularized by the Ethereum community. Typically the layer-two network periodically submits transactions which commit state updates of some kind to the layer-one network.

their Dollars for Naira instantly.[3] Meanwhile, the customer's privacy is protected – on chain, no one knows exactly who swapped with whom or how many Dollars were swapped to Naira. The market-maker knows how many Dollars and Naira were swapped, but not the identity of their counterparty.

This is the best possible privacy narrative today for payment and chat apps.

For traders, this system has significant advantages compared to centralized cryptocurrency exchanges. Your MobileCoin network tokens are always in your possession, and are never sent to any third party or smart contract. You don't have to worry that the exchange will become insolvent and you will be left holding the bag. Your off-ramp is a bridge[4] which is managed independently and with very high transparency.

Why FireDex wins

Traders are often concerned with what data is visible to who, because they want to protect their strategies and their trades. Most "decentralized exchanges" (dexes) that exist today make this impossible because everything happens transparently on-chain. All other traders can see what you are doing on Ethereum [15].

When using FireDex, things work the way Satoshi intended: Anyone can see the existing offers, but not who made them, and when you make a trade, you don't get to see who your counterparties are, because that's on a need-to-know basis.

To ensure that bad actors are kept out, on-ramps and off-ramps into and out of the ecosystem follow strict compliance rules. Bridges generally require users to be KYC'd, and exchanges that list any of these tokens are similarly obligated. Beyond this, the initial dev team has plans to extend the signed contingent input mechanism so that a trading party can require that its counterparty supplies anonymous credentials in order for the transaction to be valid.

We believe that FireDex can become an attractive and practical way for customers and traders to exchange cryptocurrency assets in a fully compliant manner and we are excited to bring DeFi to the MobileCoin network. This project will be stewarded by the FireDex DAO, an open community of supporters and stakeholders.

[3] In fact, the swap and the payment can happen in one transaction on the MobileCoin network, so there is no delay from the customer's point of view.

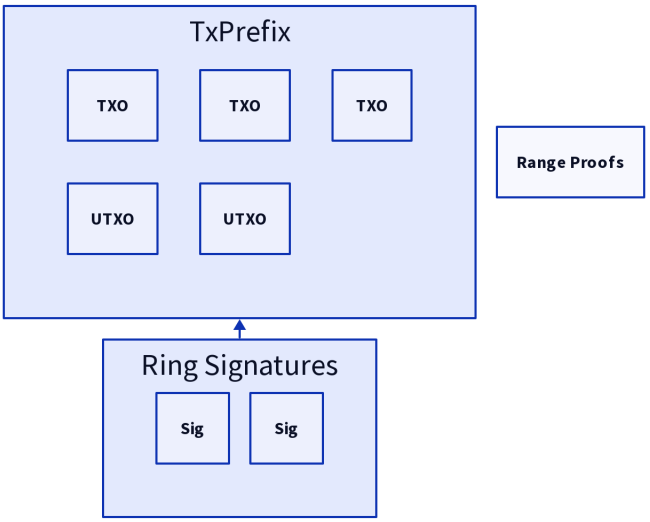
[4] A "bridge" is a system for moving tokens on one blockchain to a representation on another blockchain, and back. For more background, see references.

2 How FireDex works

2.1 Signed Contingent Input Mechanics

At a high-level, MobileCoin's blockchain is UTXO-based, similar to Bitcoin. This means that the ledger contains an immutable list of "transaction outputs" (TXO's), which represent "credits" in the system. Each transaction creates new transaction outputs, and spends earlier transaction outputs (the "inputs"), consuming them but leaving them on chain. A valid transaction can only spend "unspent transaction outputs" (UTXOs). This model has better privacy properties than "account-based" models, and so is the approach taken by most privacy blockchain projects.

Each TXO contains several elliptic curve public keys which represent the address that owns them, and uses a Pedersen commitment to represent the (encrypted) commitment to value. The elliptic curve public keys represent a so-called "stealth address", and are derived using the owner's public address, but this linkage is blinded using randomness when the transaction is built. The owner's view key is needed to identify when a TXO belongs to them.



A MobileCoin Transaction

A MobileCoin transaction has a few more elements than a Bitcoin transaction. A valid transaction includes a proof that the encrypted input values are equal to the encrypted output values, without revealing those values to the verifier.[5] Each input TXO also requires a "ring signature" from the owner, which signs over the whole transaction prefix, and establishes the owner's intent to create this transaction.[6] This transaction can then be submitted over an encrypted channel to the SGX enclave within a MobileCoin consensus validator node. For more detail, we refer the reader to "Mechanics of MobileCoin".

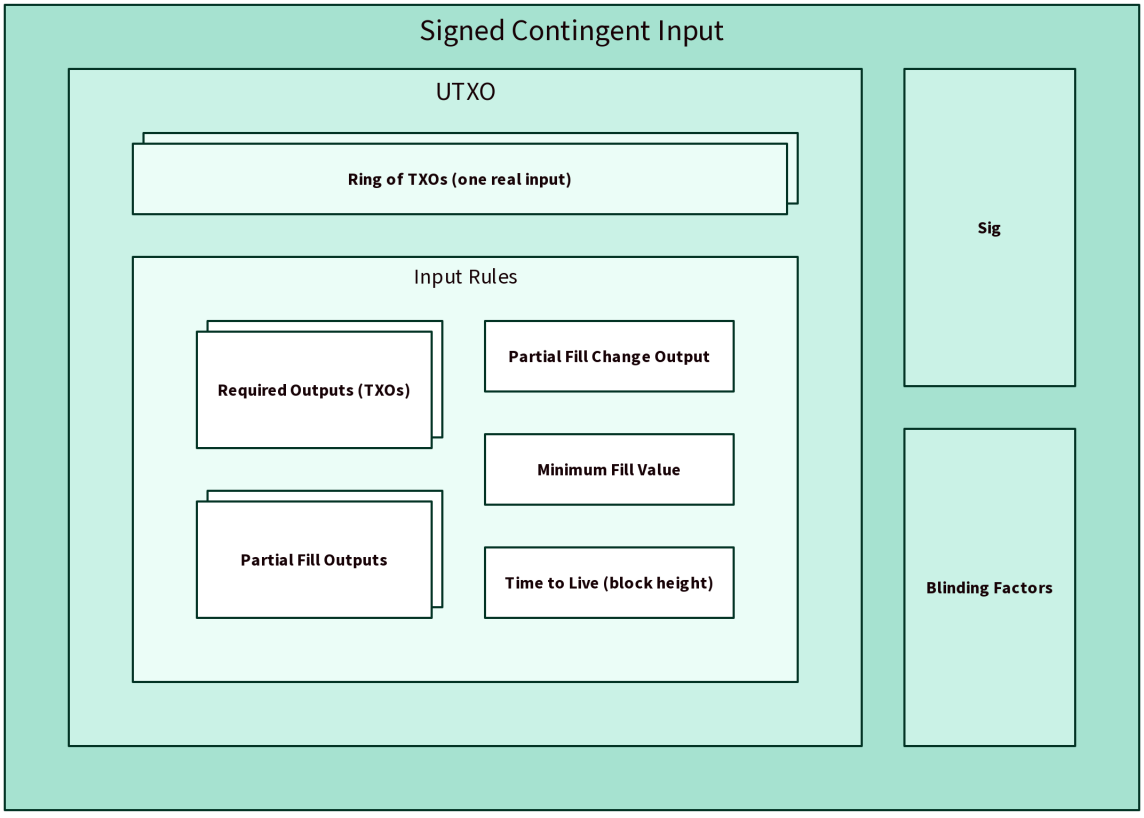
Upgrades to the MobileCoin network happen via the "MobileCoin Improvement Proposal" (MCIP) process. MCIP #31 introduced "Signed Contingent Inputs" (SCI), which gives signers more flexibility, as a means to enable peer-to-peer swaps. Within a transaction,

[5]This proof isn't necessary in Bitcoin because the TXO amounts are not encrypted, but the node checks the same equation. See also [11].

[6]In Bitcoin this is a more standard ECDSA signature.

an input now has an optional "rules" field. Rules are a list of requirements on the form of the overall transaction. Inputs with rules no longer sign over the whole transaction – the signatures for these inputs only sign over themselves, including the rules. The rules are encoded in a simple schema, which is interpreted by the enclave, and checked against the final transaction, with semantics specified in that and subsequent MCIPs.

The purpose of this is that it becomes possible for one token holder to sign away the right to their token, as long as whatever transaction this signature is used in ultimately conforms to the signer's requirements. For example, if Alice has eUSD and wants MOB, she could sign for her eUSD output in a way that requires that she is provided with a MOB output. To do this, she makes a (blinded) MOB output for herself, and creates a rule that this output must appear in the transaction. Then she signs over the eUSD input and rules using her private keys. The result is an SCI. Conceptually, this is a "transaction fragment" that someone else could use to build a complete transaction.



A Signed Contingent Input

SCIs enable a simple 2-step protocol for mutually distrusting counterparties to swap currency. In the first step, Alice builds and signs an SCI and sends it to Bob. Then, Bob builds and submits a transaction that uses the SCI. If Bob chooses not to complete the trade, nothing is lost from Alice – the funds are still in her possession and nothing changed on-chain. There is also nothing that Bob can do with the SCI other than use it to conduct the trade proposed by Alice.

Once Bob builds and submits the transaction using the SCI, each of Alice's and Bob's respective inputs will be consumed, and each of their respective outputs will be produced, in the block in which this transaction becomes final. The swap is "atomic" in that both

balances update in the same block, and there is never a point at which one party has neither the MOB nor the eUSD.

More interestingly, the SCI need not be signed with knowledge of the counterparty's identity. This means that Alice can broadcast her SCI to many potential counterparties, not just Bob. If one of them agrees with the terms of the offer, they can add the signed input to their transaction as normal, and the result will be valid as long as they produce the correct output in exchange. They can produce an eUSD output for themselves to consume the eUSD supplied by the SCI, but they will have to supply their own MOB in order to create a balanced transaction at the end. Then they can sign and submit this transaction.[7]

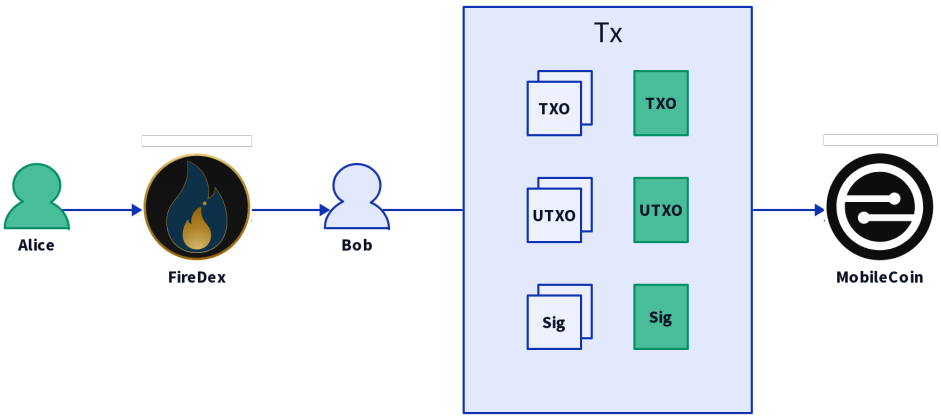
MCIP #42 extended this with partial-fill rules. This allows Alice to sign that she is willing to trade eUSD for MOB at a certain rate up to the volume of her input, and her counter-party can fill this offer to whatever fraction they like. Alice can also set minimum fill amounts to prevent dust attacks.

As FireDex matures, the initial dev team can collaborate with the MobileCoin community to extend these features via the MCIP process.

Validating a transaction that uses SCIs does not require significantly more computation than validating a normal transaction, and so there are no additional "gas costs" around transactions that settle trades using this system.

2.2 FireDex Node Mechanics

In order to build an efficient marketplace on top of this technology, FireDex proposes a layer-two network over the MobileCoin network, which broadcasts and distributes SCIs between market participants world-wide.



FireDex protocol

As a matter of strategy, this layer two network is not a consensus layer – it is a carefully designed peer-to-peer gossip network and does not require sequencing the offers before distributing them. This has the advantage that the throughput of orders is not limited to the throughput of a blockchain. Only the settlement of trades requires the blockchain.

[7]Multiple SCIs can be added to one transaction, in order to settle multiple trades simultaneously. This can be useful for making multi-step currency swaps happen in an all-or-nothing fashion.

Think of this as a peer-to-peer streaming protocol for trading.

Each FireDex node runs an instance of `mobilecoind` (the original MobileCoin wallet program), which synchronizes a copy of the blockchain ledger. This is used to automatically prune orders that have filled from its local copy of the order book.

Orders are added to the order book when they are received via an RPC endpoint, and determined to be valid. Any orders which are new are also broadcast to the node's peers. In a well-connected gossip network with honest participants, each order is quickly delivered to every peer. It's also possible for a super-majority of honest participants to statistically detect dishonest nodes that hoard or delay orders that they receive. Such dishonest nodes are then punished by cutting them out of the network.

Market participants can ask each node for the current state of its order book – it can provide summaries, or entire SCIs, which are typically about 1 kb in size, including all proofs needed to submit them to the blockchain.

2.2.1 Order-book Semantics

To make the analogue of a "limit order", a market participant simply creates an output of the appropriate size, and then signs an SCI over it, with a required output for themselves. This implies price and volume for the order, in traditional terminology. Then they submit it to (one or more) FireDex nodes, and wait for it to fill.

Typically it's expected that they'll use partial fill rules here, which means the output for themselves is a fractional output, and they'll create a fractional change output for themselves as well. Then their counterparty can decide to what fraction to fill it. But, they could also place a "fill-or-kill" type order by declining to do that.

In the case of a partial-fill limit order, it's possible that their SCI will be partially filled by a counterparty, but the input they signed is now consumed on the UTXO blockchain. In this case, if they want to continue matching, they must sign over the change output they received from this and submit again.

To place the analogue of a market order, a market participant asks (one or more) FireDex nodes for the current state of the order book. Then they build a transaction which consumes the favorable orders assuming the prices are acceptable, and submit the final transaction to the blockchain themselves to settle the trade.

Participants can use the system however they want, but what we predict is that market-makers will typically run an instance of `mobilecoind` and produce partial-fill limit orders tracking current prices. These types of users are in a position to run highly-available infrastructure to support their automated trading.

In contrast, users on phone apps will likely prefer the second approach, where they pull snapshots of the order book when they want to make a trade, and submit transactions to the blockchain to settle these trades.

- A user on a phone app typically wants a trade to execute immediately, for convenience. If they place a limit order, they won't really know when it will execute.
- A phone app can relatively easily use diverse price feeds to ensure that they are getting acceptable prices. Because the phone controls matching, the user is in a

great position to set slippage bounds and such.

- As in any market, it's possible that another user will match against the liquidity you wanted before you do. In FireDex, if this happens, the transaction fails at the consensus level, and you can simply retry. To reduce the chance of collisions among users, we expect that market-makers will typically post multiple identical partial-fill limit orders, and that phone users will choose among these randomly.

As in traditional markets, a limit order can also be cancelled. Once an SCI has been signed and broadcast, the only way to cancel it is to make it invalid at the blockchain level, by self-spending the underlying input, or letting it expire (based on a time-to-live value specified as a block-height). If an order needs to be canceled before it expires, an on-chain event is necessary. The MobileCoin network is fast and transactions are cheap, so this shouldn't pose an issue for market-makers and traders.

2.3 DAO Mechanics

The FireDex DAO treasury will be bootstrapped by selling the FIDX token, an ERC20 on the Ethereum blockchain. FIDX token holders will vote as DAO members, in proportion to their holdings. There will be a fixed supply of 1 billion FIDX tokens.

In order to further incentive development of FireDex, the FireDex nodes collect trading fees. This is achieved by requiring that all incoming SCIs have a fractional output which pays some number of basis points of the trade to the FireDex fee address, as specified by the DAO. This is similar mechanically to the MobileCoin network fee address.

All fees collected will accrue to the FireDex DAO. Initially, we expect that the FireDex DAO will appoint a steward who will aggregate fees on a monthly basis, report totals, and bridge revenue to the DAO treasury as may be desired by the DAO.

An interesting aspect of this is that, if one knows the fee revenue due to e.g. a 10 basis point trade fee on a per-trade basis, then one has enough information to compute the size of each trade, but one of the design goals is to keep such information private. It is possible for a steward to aggregate all the fees over a monthly period, and reveal the total value of all fees collected in that time, in a manner that is provable from on-chain data, without revealing the amount of any individual fee paid. In such a model, this would be the only way to determine the total volume of funds that moved on FireDex in a given period of time. In this approach, it is possible for the DAO to trust-but-verify that the steward correctly handled the fee account. On the other hand, the DAO may vote instead to simply reveal the view key of the fee account if they prefer that.

To reduce trust in the steward, one possible mechanism is to move the fee account keys into an SGX enclave, and automate its operation. This could be one possible route to fully-automated governance for the FireDex DAO, but there are other possibilities. The DAO is expected to decide upon strategy around governance mechanisms.

The FireDex DAO will vote on all policy matters around FireDex including what fee rates and fee policies would be in place, what to do with any fees that are collected, which versions of the FireDex node software to bless as canonical, what software configurations should be used when running a FireDex node, and generally decide upon the goals of the project and how best to advance them.

2.4 Distribution

There are a number of different potential ways that end users can ultimately use FireDex.

- We hope, but cannot be sure, that existing apps that have integrated MobileCoin wallets will integrate FireDex in order to offer low-cost, convenient services to their users. This may be especially helpful for international peer-to-peer payments.
- The initial dev team would like to build an open-source, browser-based MobileCoin wallet which integrates MobileCoin Fog. (See MobileCoin Fog whitepaper for background.) Such a wallet could become an easy way to trade using FireDex.
- The initial dev team is interested to create bindings for libraries like `ccxt` to place orders on FireDex using `mobilecoind`. This can help traders with existing trading bots start using FireDex easily.
- Larger market-makers may prefer to integrate `mobilecoind` directly into their existing infrastructure.

The DAO is expected to pass resolutions and help guide any developers that choose to work on these efforts.

3 Appendix

3.1 Future Directions

In the model described so far, FireDex functions roughly as a Central Limit Order Book (CLOB), except that the order book is decentralized and propagated by a peer-to-peer network, and the exchange doesn't have custody of funds. Participants can all see the order book, but don't know the identities of the various parties.

A future direction which we're interested in exploring is to use this technology to build a dark pool. In a dark pool, participants cannot see the order book. Instead, they can only submit orders, and a matching engine finds matches that result in trades. Dark pools have become increasingly popular in recent years in traditional finance, especially as a way of settling large block trades without impacting the market.

We propose to build such a matching engine inside of an SGX enclave, and for FireDex nodes to optionally run this as an extension. The matching engine can be transparently implemented in a manner that conforms to traditional exchange semantics, and can directly itself submit transactions to the MobileCoin network. SCIs would either enter such a dark pool by being imported from the public pool, or participants could submit them directly to the enclave.

For resiliency, these enclaves can peer to each other using mutual attestation, similar to how the MobileCoin network nodes do, and replicate SCIs in the dark pool securely to their peers.

Besides the dark pool use-case, such a design based on enclaves has some other benefits.

- It may promote orderly execution with less contention, since there would be fewer places where the matching process is happening simultaneously.

- When the SCI's are not publicly shared, and are only sent to enclaves, it becomes easier to cancel them (before they expire). The originator can simply sign a request for the dark pool enclave to delete it, and to broadcast this cancellation request to its peers. This doesn't require an on-chain event, so it doesn't pay a network transaction fee or compete for block-space. By contrast, when an SCI is shared publicly, one can't ever be sure that all copies were deleted, so the only way to be sure that it has been canceled is to invalidate it at the level of the chain, by spending the underlying input or similar.

3.2 Competitive Advantages

There are a number of key advantages of FireDex compared to traditional centralized and decentralized exchanges.

- FireDex is completely non-custodial, and does not even have smart contracts or "decentralized entities" that receive control of funds at any point during a trade. Funds are at all times in possession of the users.[8] This greatly reduces the risks involved in using FireDex. Many traders lost a great deal of money in FTX's collapse. Other traders have been burned by providing liquidity to smart contracts that later got hacked. For such traders, FireDex may be more attractive, because it doesn't have counterparty risk or smart contract risk. These are major differentiators.
- FireDex is one of very few DEX approaches that have comprehensive mitigation for MEV. The mitigation is very simple – signed offers are distributed off-chain, and any settlement only occurs when one party builds a transaction and submits it to MobileCoin. SGX prevents any details from being revealed until the transactions are final. Besides this, the model doesn't allow for "market orders" resolved by the blockchain – racing another user to the chain can only cause their transaction to become invalid, and can't cause them to get a worse price than expected.
- FireDex is mobile-ready from day one. By virtue of building on MobileCoin and being compatible with MobileCoin Fog, phone users can use the system without compromising on privacy or performing expensive sync operations. Currently no other privacy chain has a similar technology, and transparent blockchains that support DeFi generally suffer from MEV. FireDex checks these boxes and has a clear path to distribution. This also gives it a second target user-group, and volume from that user group may help support the first.

We can also compare and contrast FireDex with many existing solutions.

- Compared to AMMs, FireDex doesn't require liquidity pools and product formulas, which often result in a very capital intensive product when volatile assets are present. AMMs generally don't provide nearly as much liquidity to traders at the end compared to what can be done with the same capital on a CLOB, when volatile assets are traded. Uniswap V3 introduced concentrated liquidity (CLMM), which improves things somewhat, but empirically it is still more capital intensive than what we see on a traditional CLOB.

[8] In traditional terminology, FireDex is actually closer to a **quoting service** than an exchange. Possibly, it should be called FireDeqs instead to emphasize this distinction. It is primarily called a DEX here because that concept is already widely understood, and the user experience is similar.

- A number of projects are trying to do CLOB on-chain in the Solana ecosystem, notably Serum. But it seems very difficult to scale a blockchain to handle the kinds of traffic from "high-frequency traders" (HFT) that traditional centralized exchanges experience, especially if every order must happen on-chain. We think that a better approach is to try to keep order flow off-chain and only perform settlement on-chain, since there may be orders of magnitude less on-chain traffic this way, and there isn't a compelling technical reason for all order events to be on-chain. Solana seems to be the main venue for this, but Solana has the same MEV problems as Ethereum, and projects like Jito MEV bot help node operators to extract MEV auotomatically, similar to Flashbots.
- Some projects like Sei network propose using "Frequent Batched Auctions" (FBA) on-chain, rather than CLOB, as a solution to MEV. However, there are a few problems with this. One is that MEV should be properly thought of as a security issue – in traditional finance, all forms of manipulation performed by an exchange along these lines are illegal in most jurisdictions. But FBA is not a fix to the underlying security issue, it is only a change to the trading model. Miners may still find new creative ways to exploit their position to their great advantage as long as they can see the trades coming in before they are final. Another serious problem is that FBA has been proposed in the traditional financial industry for the better part of two decades, as a "solution" to HFT. Despite this, many practitioners push back on the model, and it doesn't appear that FBA exchanges have found product-market fit in traditional finance. (Penumbra also uses FBA, but not as a primary mitigation for MEV.)
- In a similar category, UniswapX proposes to "internalize MEV", capturing some of the value lost due to known MEV-boost block-builder techniques and trying to return it to liquidity providers and traders. However, it's important to note that it can only do this for some very specific forms of MEV. Smart contract techniques cannot fix MEV by themselves – it is fundamentally a security problem in the design of the blockchain.
- There are a few other interesting projects like Penumbra that propose to use threshold cryptography to try to resolve MEV. (Earlier, researchers from Osmosis and Anoma proposed using threshold cryptography for this as well and published "Ferveo" [6]. This isn't at implementation phase yet in Osmosis, but it is in Penumbra [30] and is also cited by Anoma [17].) The idea is that transactions should be secured by threshold encryption when they are submitted to the network – each node publishes a public key, and the transactions are encrypted such that e.g. 2/3 of the private keys are needed to decrypt them. The nodes perform consensus on the encrypted transactions, and then collaboratively decrypt them and implement the changes after consensus has chosen the transactions which will be in the block. For projects based in the cosmos ecosystem, this feels very natural since tendermint consensus is based on the assumption that 2/3 of the nodes are honest anyways. However, in practice there are good reasons to doubt that this will work.
  - There are numerous small proof-of-stake networks today, which very rarely have consensus failures. But the best way to explain this is not that "most people are honest" or "it is hard to get 2/3 of people to collude". Indeed, most of these networks are run by small groups that know each other well, and

have already collaborated well enough to run a nontrivial distributed system together. The best way to explain why these networks work in practice is that if the validators colluded to undermine consensus, it would be impossible to conceal this from the users. Exchanges that listed their project would find out, and if they lost funds in a rollback attack, would likely delist them. Any time a consensus failure or unexpected hard-fork happens, it is controversial and negative for the project as a whole. In other words, the users are in a position to trust-but-verify that the validators behaved honestly, and the validators collectively have immediate negative consequences if they collude maliciously. When using threshold-cryptography and 2/3-honesty assumptions for *privacy*, this element of trust-but-verify is lost. If the system is based on the idea that 2/3 of the nodes will not collude to decrypt the transactions early, and then re-order or front-run them, the problem is that if the node operators did that, it would be extremely difficult for anyone to find out. This completely changes the game theory of whether they will actually collude. If the total value of MEV on Ethereum network is any indication, it will be extremely lucrative for validators to collude and break threshold cryptography if such systems attract significant volume.

  - Hardening such a system with secure hardware, such as SGX, therefore seems wise. Secure hardware sometimes has bugs, but it is unlikely that a company like Intel would collude to undermine their own technology to help validators attack the system for nefarious purposes. SGX has been used successfully in practice for years by many serious projects including Signal messenger, OnePassword, and Fortanix, to name a few. Possibly the single highest-value application is the Ava labs bridge, which has secured billions of dollars of cross-chain cryptocurrency transfers using a signing key in an SGX enclave.

- When developing privacy-enhancing technologies, it's very important that you produce a very clear model of what is private about the system that users can understand. Many engineers naively believe that adding noise, or using obfuscation techniques, is good enough in practice. A famous talk by Ian Miers entitled "Satoshi has no clothes" [23] discussed the pitfalls of this, particularly comparing "decoy-based" systems like Monero with zero-knowledge proof-based systems like ZCash. When privacy is based on heuristics or guess-work ("there will usually be a lot of transactions in the block"), it turns into a pitfall for the users ("do I still have privacy if I'm the only person transacting in this block? How can I ensure that I'm not? I guess I can't."). Or, attackers may be able to undermine the assumptions actively, for example with dust attacks. Then, if someone actually needs to rely on the system in a high-stakes way, it can become very risky for them.

  - FireDex's design enables very clear guarantees for the user, which is a significant advantage. For example, if a user pulls the order book, chooses an SCI, and submits a transaction, they can be sure that no one besides their counterparty will learn to what degree they partially-filled this order, because the enclave doesn't reveal that and the TXOs that result are encrypted. The SCI will become invalid after the next block, and it will be clear that a partial-fill output that it required was produced, so it can be inferred after-the-fact that someone matched this order. The maximum volume of that order is public

knowledge, but there's no other part of the system that reveals information about the actual volume. A user could even choose to match against a higher volume partial-fill order, even if the price is worse, in order to reveal less information this way. Additionally, as the phone user pulling the order book, there's nothing on-chain or in the order book that links your new outputs to your previous inputs.

  - By contrast, in systems like Penumbra, the total trade volume in each pair for the block has to get revealed to the system so that the matching engine can run on clear-text values. But, this is not zero-knowledge with respect to your trade volume. If you're the only person in the block, your trade volume is revealed. If the only other person in the block is Eve the Eavesdropper, Eve still figures out your total trade volume. The project recommends to split your trades into many smaller trades randomly, but this is an obfuscation technique, and may not work in the face of traffic analysis or other side-channels. It's hard for you as a user to use the system in such a way that any of the details of your trade (what you traded, at what volume and price) are actually sure to be concealed, and won't be inferred exactly or approximately by an adversary.
  - The system may rely on SGX, but from a user's point of view, that is an assumption that can be understood in a practical sense and evaluated in context, just as well as an unproven knowledge-of-exponent assumption used in a ZK-SNARK.

- FireDex also has practical advantages over privacy projects such as Penumbra in that to actually use Penumbra with any privacy, you need to run a full node yourself, which downloads and scans the shielded pool. Otherwise, you are sharing most of your information with the RPC endpoint. Realistically, your average person only wants to use it in a browser extension or in a phone, and isn't going to run a full-node or anything close to it [25]. But if they don't, it's not clear what value they actually get from using Penumbra. Penumbra could adopt a technology like MobileCoin Fog [9] so that browser extensions and phones can quickly and privately find their on-chain private data without syncing the whole chain. But they would have to allow secure hardware into their threat model in order to do this. There aren't any known techniques for this that scale without using something like SGX, despite many years of research.
- Other dexes based on sharing quotes off-chain include AirSwap (on Ethereum network), and Arcane Finance (on Aleo testnet). However, these systems only provide an RFQ API. They don't actually create a shared order book and the quotes typically require additional rounds of interaction to result in a trade. Intuitively, in an RFQ system users are negotiating the prices directly with a handful of market-makers over a bilateral communication channel. This is less efficient from the user's point of view than an open, public marketplace, and results in less favorable pricing and volume. (Arcane Finance also provides AMMs on Aleo testnet, but it works in the same manner as Uniswap V2 on Ethereum. None of the privacy features of Aleo can be used here to improve the system or prevent MEV, because an AMM requires global shared state. Updates to this shared state are necessarily visible to anyone, and so any transaction that trades with the AMM reveals its impact on the AMM pool balance before it is executed.)

- Another interesting project in this space is `enclave.market`, backed by Ava labs. This project creates a full exchange and matching engine inside of an SGX enclave, which settles trades on-chain [3]. The main disadvantage of their approach is that to use the platform, you have to give custody of your funds to the enclave, so you have to compromise on self-custody. This raises the stakes dramatically on the security of the SGX enclave. It also may or may not ultimately have regulatory implications, since the enclave has custody and control of funds in some sense. This approach also adds more friction to the use of the system for currency exchange during peer-to-peer and point-of-sale transactions, which are a target use-cases for FireDex. Overall, it is still a categorical improvement over traditional centralized exchanges by many criteria.

Lastly, we consider that our compliance experience is a significant advantage. Many privacy blockchain projects either don't ever get any volume, or fail to navigate compliance effectively and get shut down. From our earlier work on the MobileCoin network, we are very experienced in navigating the delicate balance between the need for user privacy and compliance with the law. We can't control the DAO, but we can help it to achieve similar success.

For example, anonymous credentials are a great way for users to prove to eachother that they are KYC'd without revealing their identity. We are interested in using this technology to allow FireDex users to require that their counterparties have completed KYC with one of several entities of their choice, and making this technology easy to use in practice. (Such choices can be made by the users rather than the platform.) This may help more users choose to use FireDex.

References

[1] Hayden Adams, Noah Zinsmeister, Moody Salem, River Keefer, and Dan Robinson. Uniswap v3 whitepaper. uniswap.org, 2021. `https://uniswap.org/whitepaper-v3.pdf`.

[2] Sunny Aggarwal. Interchain conversations ii - threshold decrypted transactions: Thwarting front-running and enabling privacy at the protocol level. Interchain Conversations youtube stream, 2021. `https://www.youtube.com/watch?v=6WrFlsDSUYg`.

[3] Nina Bambysheva. Enclave markets adds spot trading platform with privacy features. Forbes, 2023. `https://www.forbes.com/sites/digital-assets/2023/05/03/enclave-markets-adds-spot-trading-platform/`.

[4] Simon Barber, Xavier Boyen, Elaine Shi, and Ersin Uzun. Bitter to better - how to make bitcoin a better currency. In Angelos D. Keromytis, editor, *Financial Cryptography and Data Security - 16th International Conference, FC 2012, Kralendijk, Bonaire, February 27 - March 2, 2012, Revised Selected Papers*, volume 7397 of *Lecture Notes in Computer Science*, pages 399–414. Springer, 2012.

[5] Mikolaj Barczentewicz. Mev on ethereum: A policy analysis. ICLE White Paper 2023-01-23, 2023. `https://ssrn.com/abstract=4332703` or `http://dx.doi.org/10.2139/ssrn.4332703`.

[6] Joseph Bebel and Dev Ojha. Ferveo: Threshold decryption for mempool privacy in BFT networks. *IACR Cryptol. ePrint Arch.*, page 898, 2022.

[7] Chris Beck. Mcip #31: Transactions with contingent inputs. MCIP process, 2022. https://github.com/mobilecoinfoundation/mcips/pull/31.

[8] Chris Beck. Mcip #42: Partial fill rules. MCIP process, 2022. https://github.com/mobilecoinfoundation/mcips/pull/42.

[9] Chris Beck. Mobilecoin fog whitepaper. github, 2022. https://www.sentz.com/learn/mobilecoin-fog.

[10] Sean Bowe, Alessandro Chiesa, Matthew Green, Ian Miers, Pratyush Mishra, and Howard Wu. Zexe: Enabling decentralized private computation. *IACR Cryptol. ePrint Arch.*, page 962, 2018.

[11] Benedikt Bünz, Jonathan Bootle, Dan Boneh, Andrew Poelstra, Pieter Wuille, and Greg Maxwell. Bulletproofs: Short proofs for confidential transactions and more. Cryptology ePrint Archive, Report 2017/1066, 2017. https://eprint.iacr.org/2017/1066.

[12] Frieder Paape Chris Hager. Flashbots: Block building inside sgx. blog, 2023. https://writings.flashbots.net/block-building-inside-sgx.

[13] Victor Costan, Ilia A. Lebedev, and Srinivas Devadas. Secure processors part I: background, taxonomy for secure enclaves and intel SGX architecture. *Foundations and Trends in Electronic Design Automation*, 11(1-2):1–248, 2017.

[14] Philip Daian, Steven Goldfeder, Tyler Kell, Yunqi Li, Xueyuan Zhao, Iddo Bentov, Lorenz Breidenbach, and Ari Juels. Flash boys 2.0: Frontrunning in decentralized exchanges, miner extractable value, and consensus instability. In *SP*, pages 910–927. IEEE, 2020.

[15] Georgios Konstantopoulos Dan Robinson. Ethereum is a dark forest. Paradigm Blog, 2020. https://www.paradigm.xyz/2020/08/ethereum-is-a-dark-forest.

[16] Henry DeValence. Penumbra: Building a private dex with zkps and threshold cryptography. Devcon, 2022. https://archive.devcon.org/archive/watch/6/penumbra-building-a-private-dex-with-zkps-and-threshold-cryptography/.

[17] Heliax. Anoma whiteapper. github, 2022. https://github.com/anoma/whitepaper/blob/6e7bf45a8fc9964fc46f516fab4073e48993d478/whitepaper.pdf.

[18] Henry Holtzman and Chris Beck. Electronic dollars (eusd) whitepaper. github, 2022. https://www.sentz.com/learn/electronic-dollars-eusd.

[19] josojo. Mev capturing amm (mcamm). ETH research, 2022. https://ethresear.ch/t/mev-capturing-amm-mcamm/13336.

[20] Michael Kaplan, Bernard Wong, Grant Haywood, and Conor Leary. Avalanche bridge: Secure cross-chain asset transfers using intel sgx. medium, 2021. https://medium.com/avalancheavax/avalanche-bridge-secure-cross-chain-asset-transfers-using-intel-sgx-b04f5a4c7ad1.

[21] Koe. Mechanics of mobilecoin. github, 2021. https://www.sentz.com/learn/mechanics-of-mobilecoin.

[22] Sei Labs. Sei: The layer 1 for trading. github, 2023. https://github.com/sei-protocol/sei-chain/blob/main/whitepaper/Sei_Whitepaper.pdf.

[23] Ian Miers. Satoshi has no clothes: Failures in on-chain privacy. Devcon4, 2019. https://www.youtube.com/watch?v=9s3EbSKDA3o.

[24] moxie0. Technology preview: Private contact discovery for signal. Signal Blog, 2017. https://signal.org/blog/private-contact-discovery/.

[25] moxie0. My first impressions of web3. Personal Blog, 2022. https://moxie.org/2022/01/07/web3-first-impressions.html.

[26] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. bitcoin.org, 2008. https://bitcoin.org/bitcoin.pdf.

[27] Shen Noether. Ring signature confidential transactions for monero. Cryptology ePrint Archive, Report 2015/1098, 2015. https://eprint.iacr.org/2015/1098.

[28] Alex Obadia. Flashbots: Frontrunning the mev crisis. ETH research, 2020. https://ethresear.ch/t/flashbots-frontrunning-the-mev-crisis/8251, also https://medium.com/flashbots/frontrunning-the-mev-crisis-40629a613752.

[29] Michael Oved and Don Mosites. Swap: A peer-to-peer protocol for trading ethereum tokens. swap.tech, 2017. https://whitepaper.io/document/165/airswap-whitepaper.

[30] Penumbra. Penumbra's dex arrives from the future. Penumbra Blog, 2023. https://penumbra.zone/blog/dex-arrives-from-the-future/.

[31] Emil Pepil. Arcane finance - dex on aleo. medium, 2024. https://medium.com/@emilpepil/arcane-finance-dex-on-aleo-bef686f1260d.

[32] Antoine Rondelet and Quintus Kilbourn. Mempool privacy: An economic perspective, 2023. https://arxiv.org/abs/2307.10878.

[33] Sajin Sasy, Sergey Gorbunov, and Christopher W. Fletcher. Zerotrace : Oblivious memory primitives from intel SGX. In *25th Annual Network and Distributed System Security Symposium, NDSS 2018, San Diego, California, USA, February 18-21, 2018*. The Internet Society, 2018.

[34] Jerry Sun. Osmosis: Diffusing liquidity across the cosmos ecosystem. Messari Research Report, 2022. https://messari.io/report/osmosis-diffusing-liquidity-across-the-cosmos-ecosystem.