CRYPTO
PRAGMATIST
PRO

We are writing this piece on the assumption that if you are still around and still reading these, you are not only interested in the key narratives that are driving innovation on Ethereum, but also how they **actually work**.

Why are certain protocols being built, how are they involved with the rest of DeFi, what are the actual mechanisms that make these innovations possible? These are all questions we consider on a daily basis, and have considered strongly recently when looking at MEV.

As such, today's breakdown tries to paint a *broad picture* of MEV, which can be a lot. But give us a little wiggle room and you should walk out of this with a significantly better understanding of the MEV ecosystem and where it's heading.

And if we can't do the job, then we will surely include the resources that will get you across the line.

Necessary Curriculum for MEV

Before breaking into the details of MEV, we need to go over some of the basic fundamentals of the Ethereum network as a blockchain. For all purposes, go ahead and skip through to find the more detailed parts on current implementations of MEV.

The reason we are deciding to include such detail is because of how difficult it has been for *ourselves* to wrap our heads around the idea of MEV–and we spent a ton of time doing so. We only began to add some color to our own understanding when fully comprehending all components.

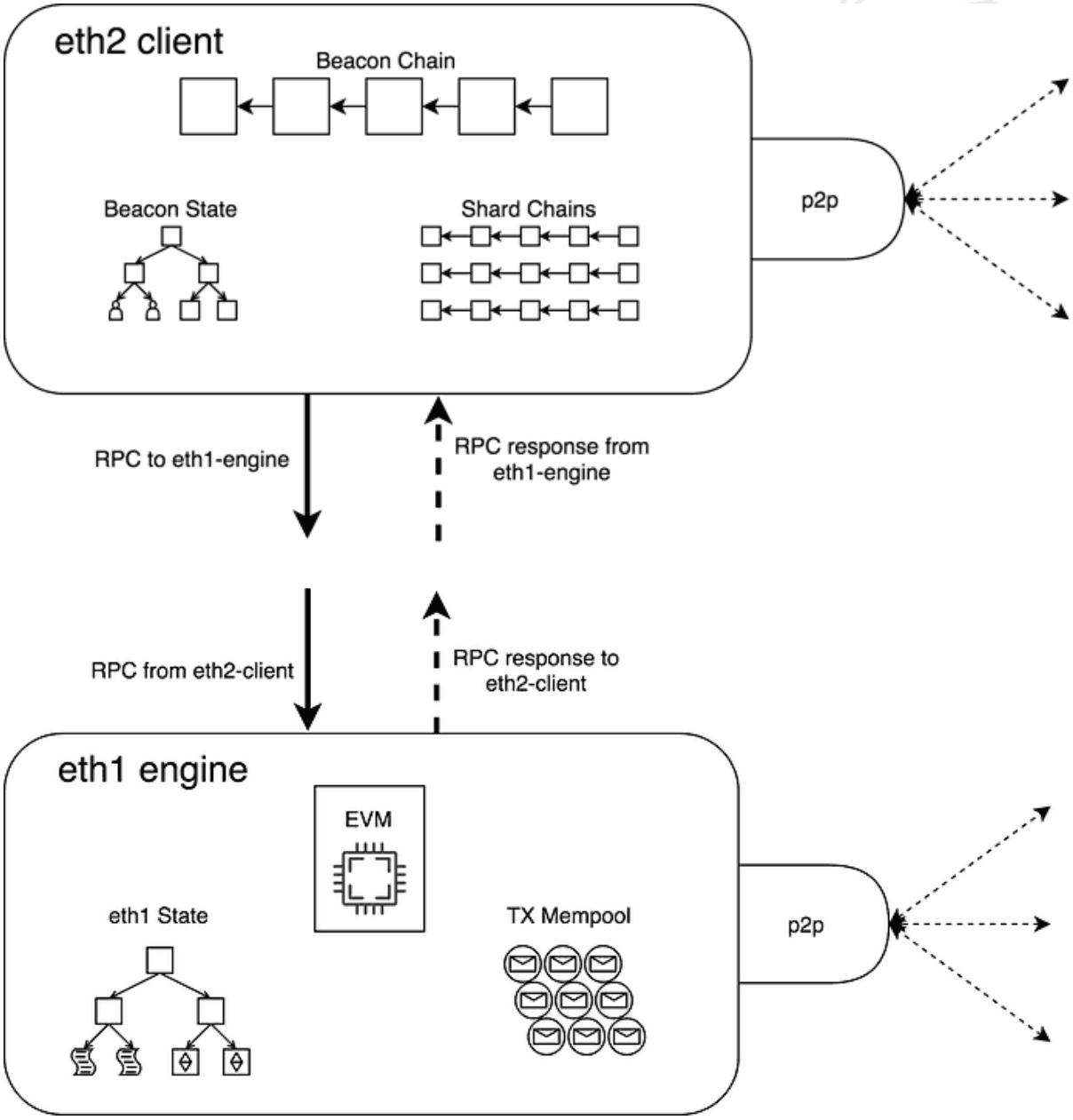
Ethereum PoS Structure

As Ethereum became a PoS chain, we know that the Beacon chain merged with Ethereum mainnet. But what exactly did this mean?

In short, **the execution clients** (previous mainnet clients) shifted from being both the consensus *and* execution layers (meaning, executing the transactions and verifying the state of the chain), to only executing and validating the block's state.

*A **Client** is simply a software application that implements with Ethereum and communicates over the network. They can read blocks and run smart contracts; the clients are what make everything on the blockchain possible.*

So, there is now a separation of the **execution** and **consensus clients**, consensus being the validators who take all of the transactions reported by execution clients and put them into blocks.



Ethereum Docs

To get further into the weeds would be a discussion regarding client diversity. This is beyond our necessary learning, but here is a fantastic paper detailing exactly why Ethereum consensus and execution is so important.

Metamask, Infura, and RPC's

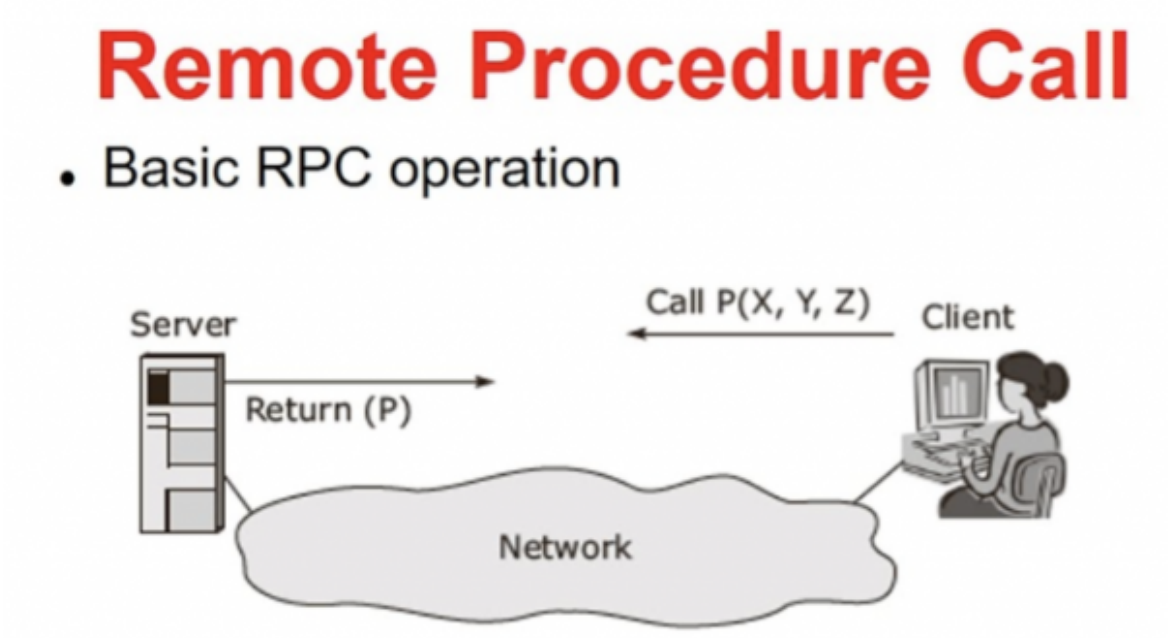
Unless we are running our own validator node or creating a DApp to interact with Ethereum, most of us will *not* be running our own client to connect with and interact with Ethereum. This is the first bottleneck, and allows for a service like Infura to step in.

Infura offers a handful of services, but the most notable is its API gateway that allows users or applications to interact with Ethereum. Infura has direct access to the blockchain, and allows anyone to use it as a “road” of sorts to access it as well.

As you might have guessed or already know, Metamask is one of the largest users of the Infura API. Given that there were more than 21 million Metamask monthly active users this summer, that is **a lot** of people relying on Infura’s service.

Metamask and Infura, however, aren’t the same service. Metamask, as we know, is our access to Ethereum. But behind that, Metamask relies on Infura for *its own access to the blockchain.* This is facilitated via **Remote Procedure Calls**, or RPCs.These are the gateways in which users (you and me) can interact with a client to then interact with the network.

Let’s think of them as a postal service for different networks; something that passes messages between two servers. The “RPC” is how the messages and information are formatted and transferred between the two different networks.



Moralis

There are two key actors with an RPC:

1. **Client** - The requesting program (sender)
2. **Server** - The remote program (receiver)

RPC’s can face problems, however, in becoming clogged with data inducing slow network speeds, incorrect blockchain data, and unreliable uptime.

In the case of Metamask, Infura is the default RPC endpoint provider. Given that there is likely a large majority of MM users sticking with the default endpoint, this can get quite clogged (and has in the past). There are two alternatives to avoiding this:

1. Use another endpoint like Pocket network or Alchemy, which are popular RPC alternatives.
2. Creating a private RPC endpoint (anyone can do this!)

Here is a simple how-to guide for using something like Alchemy (highly recommended)

These “endpoint providers” are really the connection to the Ethereum clients, which as we know are our direct contact with Ethereum. So in switching to, say, Alchemy, we are using Alchemy’s client access rather than Infura. RPC’s become very important when looking at how MEV is captured.

If any of this was too confusing and you would like another perspective, or want to dive deeper into a more technical piece, here is The Hitchhiker’s Guide to Ethereum written by Jon Charbonneau of Delphi. Okay, moving more into the good stuff now.

The Beginnings of MEV and Flashbots

So when we send transactions through Metamask (thus Infura), our transactions go through somewhat of a holding period. This is known as the “public mempool”. These are transactions that are waiting to get validated and finalized (previously, mined).

In this mempool live giga-brain super coders, since reduced to the noun of “searchers”, that find a whole bunch of opportunities to make money. Sandwich attacks, reorgs, just-in-time (JIT) liquidity are just a few ways that these programmers extract value from the everyday Ethereum user. Here is a simple thread that discusses these different types.

Luganodes @luganodes · Sep 26

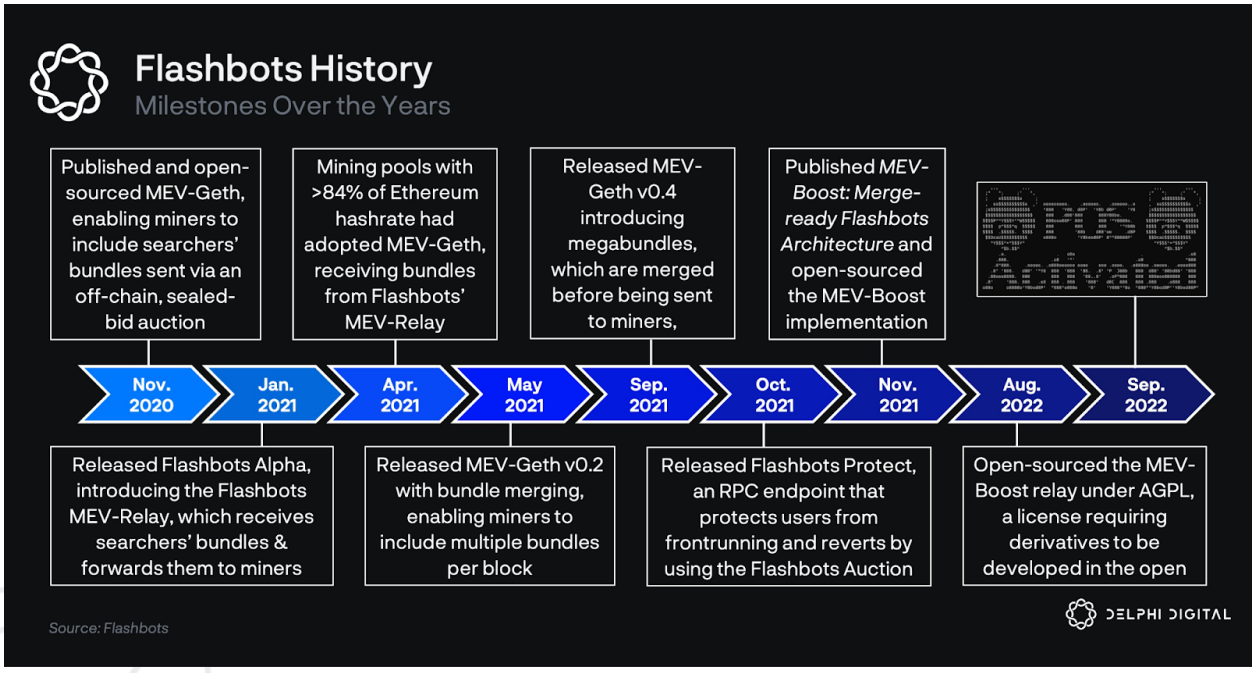
MEV Strategies: An introduction to the world of Maximal/Miner Extractable Value... 🧵👇

A thread that contains:

Introduction to MEVs
Different Types of MEV Strategies
Good Side of MEVs
Bad Side of MEVs

[Twitter]

This is what gave rise to the initial forms of MEV and led to Flashbots developing the first projects that service this “industry” within crypto:



[Delphi Digital, Jon Charbonneau]

This details the development of Flashbots MEV in the Ethereum ecosystem. What we can highlight at the end of 2020 and early 2021 are key in how things are structured today, namely the idea of **Proposer Builder Separation** (PBS), which we will discuss below.

Back in PoW days, mining was a highly concentrated industry with a small number of pools owning the largest share of mining power (sounds similar to today innit?). The first implementation of a Flashbots relay would take searcher bundles and send them over to miners. If there was more room in the block, miners would just fill in transactions that gave them the highest revenue.

This had two major implications:

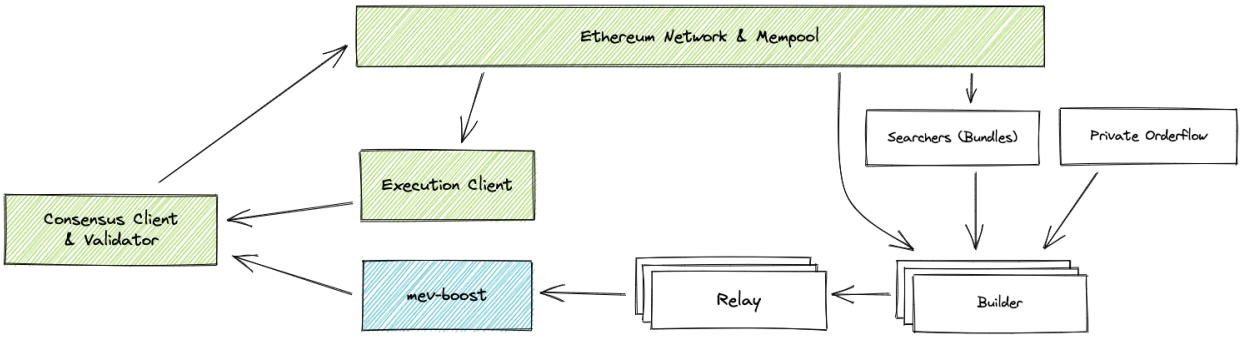
1) **Transaction censorship was effectively eliminated**. If Flashbots (or any searcher) wouldn't accept or build bundles with specific transactions in them, the miner could just append them to the block.
2) **Miners could see the contents of the bundle**, which means they could just copy and paste the transactions, but with their own address instead (keeping *all* profits). But because miner competition was fragmented, Flashbots could just stop sending them bundles, making them miss out on MEV upside.

Even in this first iteration of MEV-relay, Flashbot builders didn't include any OFAC transactions. This was not as much of an issue that we see today, however, because miners were the final builders of a block, and had the ability to append the transactions they wanted. Meaning that all transactions would *eventually* be included **as long as they paid enough in gas**.

The second point listed above, however, *was (and is)* a concern for the decentralization of network validators. Mining power, and MEV profits, was concentrated with trusted parties, which means that small scale validators would not be able to tap into this MEV market because Flashbots wouldn't trust them to send these profitable blocks.

And this doesn't stop with PoW Ethereum. PoS Ethereum prides itself on allowing anyone to spin up a node and become a validator given they meet the 32 $ETH stake requirement. But, without being a trusted party (one of the biggest validators), then the MEV market is fragmented as well.

This gave rise to the idea of PBS–which aims to democratize the access to MEV marketplace.



Flashbots

With this implementation, now people like you and me can find our way in the MEV marketplace and earn some of the upside as a validator. This is what encourages solo-staking on Etheruem, and thus better decentralization of validators.

PBS, Where MEV Stands Today

Firstly, protocol builder separation is not currently implemented at the *protocol level.* All this means is that proposers/validators (synonymous terms) are able to build their own blocks if they choose to do so.

Down the road, the Ethereum network will look to implement this at a protocol level, which would change the way consensus is achieved. There is continued research and discussion focusing on how exactly this will look (here is an example of some of the questions being raised).

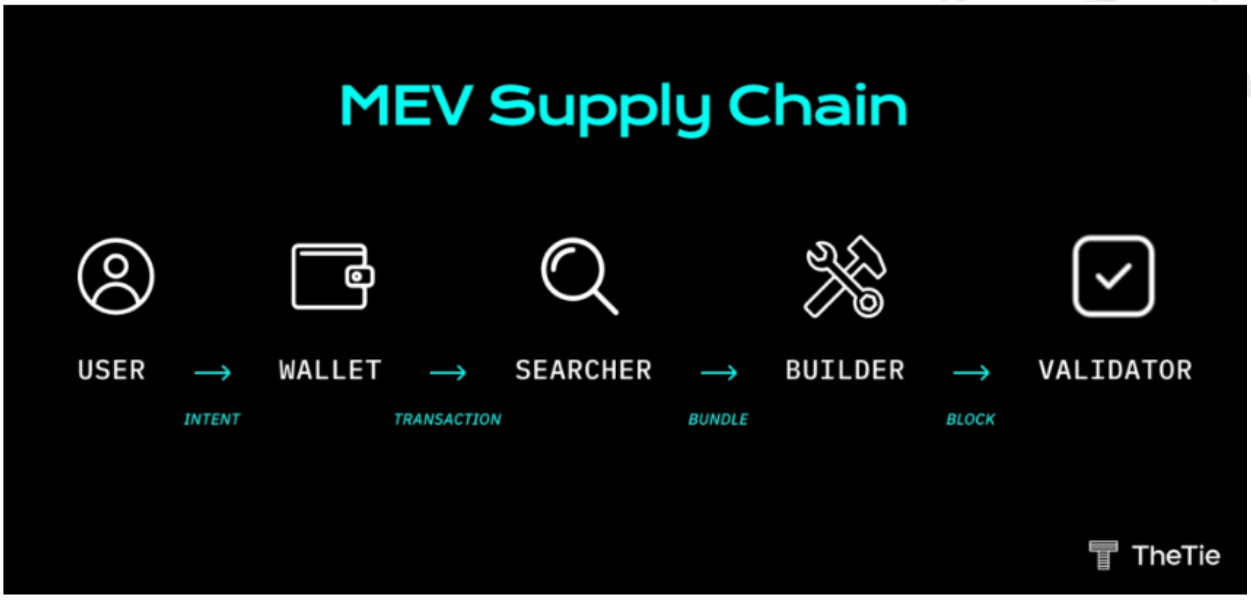
If a validator decides they don't want to build their own blocks, they can opt into the MEV marketplace that we mentioned above by using a relayer, the most popular currently being MEV-Boost.

Relayers are thus the current implementation of PBS, and facilitate blocks between the "firewall" that separates builders and proposers. In general, we know how the proposer side of things works. Let's dive a bit more into the building aspect.

Builders, Searchers, Bundles, and Relays

All of this language can become so exhausting when reading different articles and documentation about MEV. Hopefully, we will be able to abstract away some of these complications, and provide a simple framework so you can approach the entire industry of MEV with confidence and competence.

The roles we are discussing in this section have become known as the "MEV supply chain":



TheTie

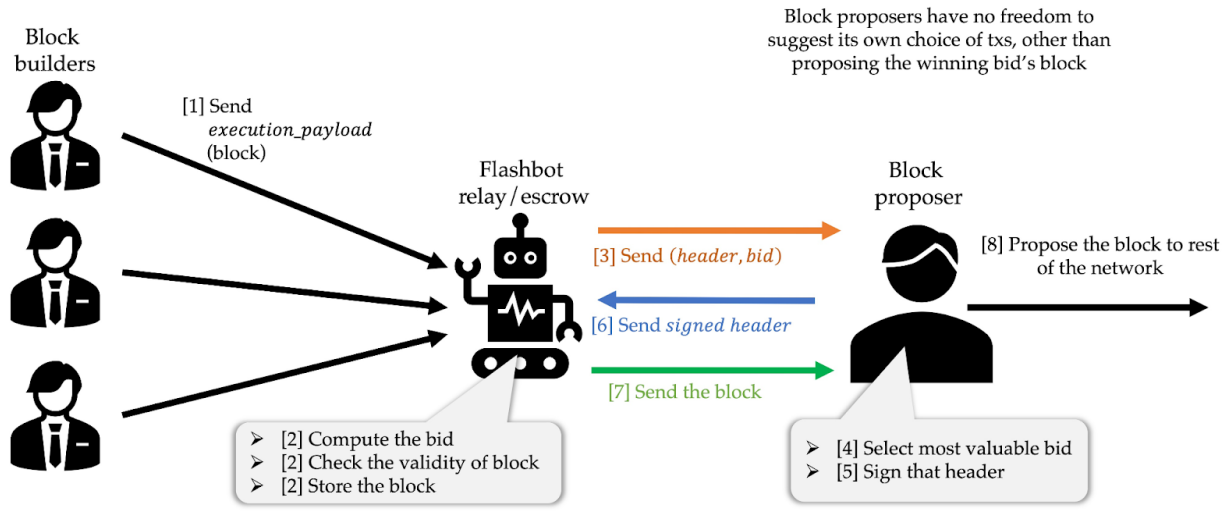
Please reference the above graphic taken from Flashbot documentation that includes the Ethereum network and mempool in addition to this one, as it will likely help tie all of these concepts together.

Searchers: This is the real technical part of MEV. The “dark forest” of transactions, and complex algorithms that run at extremely high frequencies. The searchers take transactions from wherever they can–whether that be private order flows or the public mempool (shoutout our RPC knowledge from above), and reorder these/insert new transactions to create the most profitable bundle possible. These bundles are shipped off to builders.

Builders: These can simply be an extension of the searchers, or an independent party. Builders take all of the bundles sent their way and pair them with, again, transactions from private or public mempools to create **an entire block** that is **MEV profitable.** The builder then submits this block via a block header to the relay with a “bid”.

Relayer - This can also be the “builder”, or just a separate party. The relayer is a **trusted party by the validators**. Why does the relayer have to be a trusted party?

We can think of the relayer as being involved with both block production and validation–the intermediary that eliminates trust assumptions between the builder and proposer. If that sounds confusing, here is an helpful image from EigenLayr:



EigenLayr

If that looks confusing, here's the gist.

a. The relayer shows **only the header of** all the blocks/bids given by the builders to the proposer. The proposer cannot see the contents of the block.
b. The proposer then signs the header of the block they want to choose (likely the block with the highest bid).
c. Once the block is signed, *then* the proposer can see the contents.

Once the block is signed and attested by the rest of the validators, then the profits from this whole process are distributed to the validator in the form of the transaction fee bid + MEV profits, which results in the boosted yield:

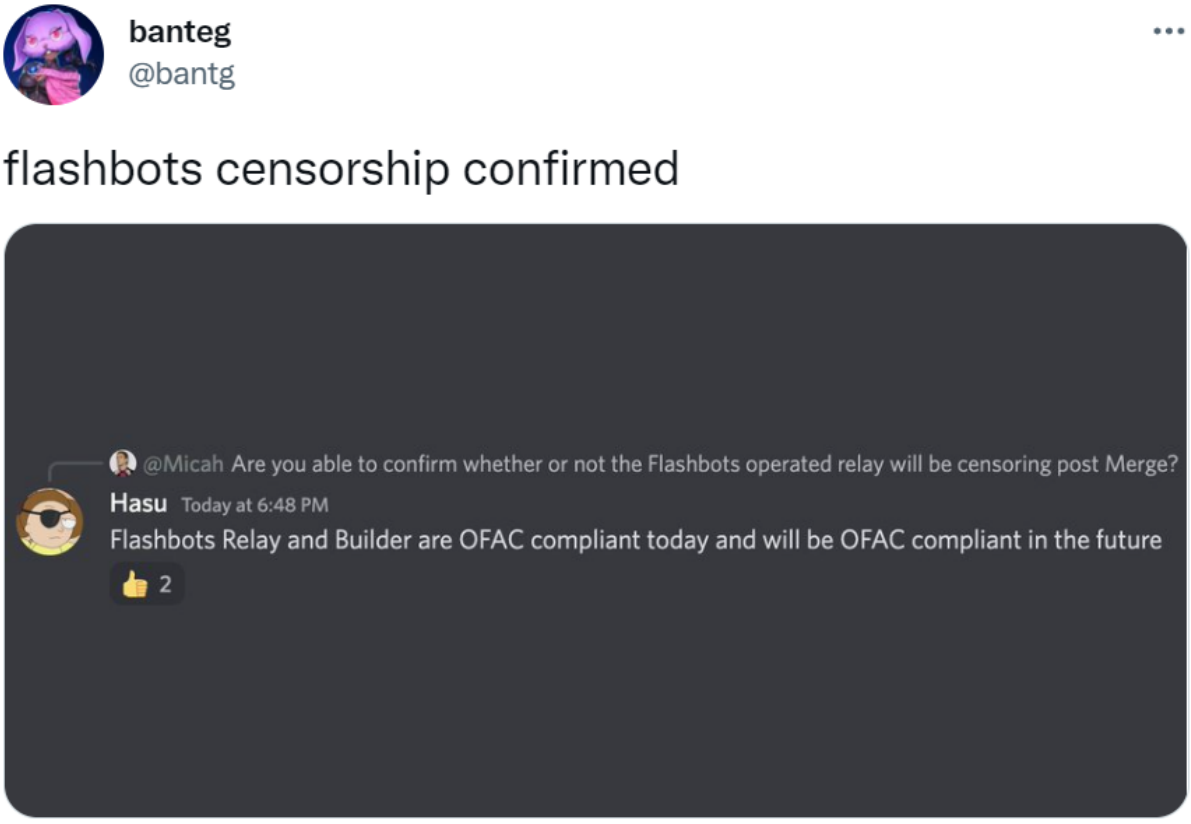
Hasu

This is referred to as a “commit reveal scheme” and helps prevent proposers from just stealing the MEV. Once a block is signed and proposed, switching to a new block (and running the MEV transactions themselves) would mean the proposer signs two blocks and **gets slashed**. Similar to how the trusted miners in PoW prevented them from doing the same. While this is necessary, it introduces a level of censorship.

Censorship is Back?

With the first iteration of MEV-Relay (PoW version), transaction censorship was not a problem because miners could include any transaction as long as it paid sufficient gas. Well, with the implementation of MEV-Boost and other PBS relays, achieving decentralization of validators and the MEV market came at the expense of transaction censorship.

This is where you have seen a lot of the rumblings on Twitter regarding Flashbots building OFAC compliant blocks–which just means they are not producing blocks that include Tornado Cash transactions:



Bantg

The OFAC is a United States government agency that sanctioned the Tornado Cash mixer contract. What followed was a pretty massive shift in the DeFi ecosystem: major protocols like Uniswap, Balancer, and Aave followed suit and issued a front end blacklist for addresses that had interacted with the protocol (you could still interact with the contract). As did USDC stablecoin issuer, Circle, which had a rippling effect around DeFi.

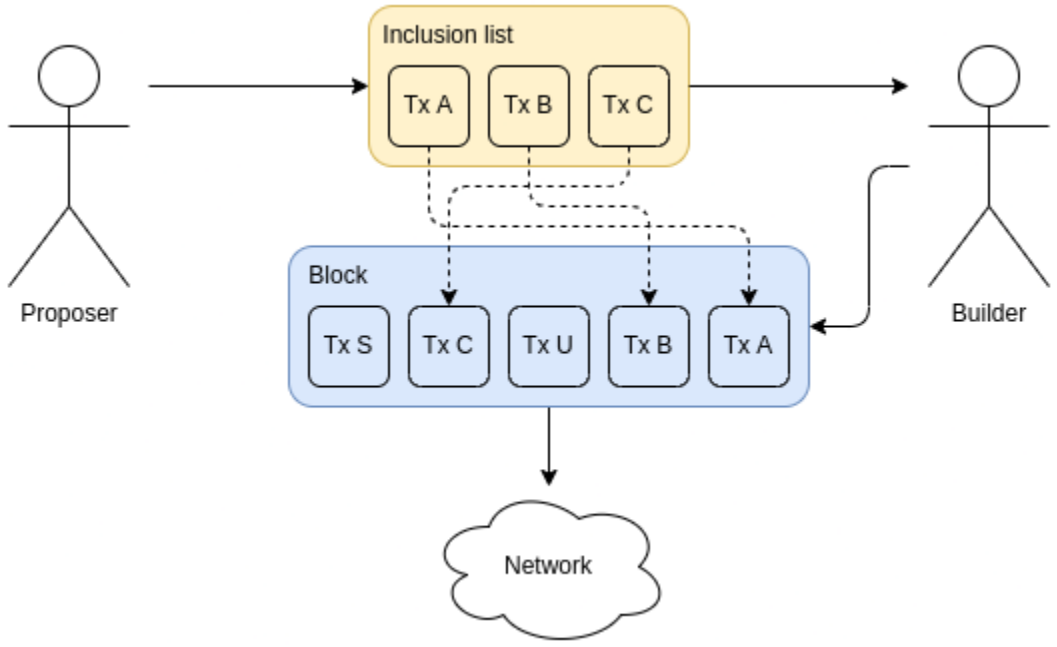
Flashbots following these compliance rules simply means they, **as a MEV builder/relayer,** will not allow transactions that have interacted with the Tornado Cash smart contract into their blocks. Even though this is absolutely a concerning and very relevant decision by the Flashbots team, they quickly made a decision to open source the Flashbots Relay, which in theory leads to more competition and options that *do not* censor transactions.

Why is This The Case?
The reason this became a problem under this new structure of PBS via relayers is because validators have to accept blocks in their *entirety.* A validator signing the header of a given block means they accept whatever transactions are within that block–OFAC compliant or not. No more appending transactions.

There are a couple of potential solutions being looked at for this, and now is where I will give a well-deserved nod to our good friend Smol Thots who wrote a fantastic piece aggregating some of the most important considerations of MEV.

Fighting Censorship and the Flaws of PBS

Building blocks is an inherently centralized process. Due to the extremely high technical know-how, hardware costs, and potential colocation, block building becomes something that is The first method (proposed by Vitalik) that makes a lot of sense is censorship resistant lists (CRlists), or inclusion lists. These are lists specified by the validator of transactions they want to include in the given block.



Ethresear.ch

While this does help solve the issue of censorship by allowing a proposer to include any transaction they want via the builder, there are some basic flaws:

1) The proposer would have to have the ability to read the state, which increases the requirements to run a node, hurting decentralization
2) Proposers could just write empty lists to ensure all builders will always be willing to build blocks for them

For the sake of word count, that is the only one that we will discuss today as these solutions can get pretty complex. Again, give the paper by Smol Thots a read for more details and links to different research pieces on these solutions.

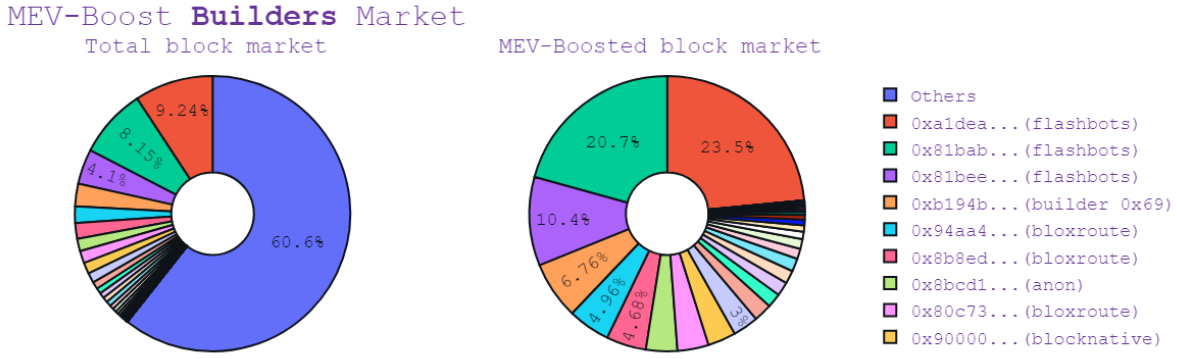
MEV Chain of Value Capture:
Some of this was either assumed or developed through the last section, but here we will detail exactly who is making money and *how* they are making money in the process of MEV.

When searchers build a bundle, they send it off to a builder with “priority fee”, or a tip that incentivizes a builder to include it in their construction of a block. Remember, a builder is economically incentivized, so any searcher that sends a high tip will get their bundle included. The builder then creates their most profitable block, and sends the header of this block, with a

Current MEV Market: MEV-Boost, Flashbots, Manifold and More
Note that data in this section has been under debate by various parties involved in block building. In general, the MEV market is one of the most nascent, yet impactful, markets on Ethereum, so data accuracy is crucial. Thus, the data presented are informational only and should not be relied upon for investment decisions.

Obviously the biggest talk of MEV currently is Flashbots and their MEV-Boost relay. Currently, they are dominating both in the building and relay market:

- MEV-Boosted blocks account for roughly 40% of blocks built on Ethereum
- Flashbots builders currently producing 54.6% of those blocks



Mevboost.pics

As Flashbots build OFAC compliant blocks and commands such a large share of the MEV market, people are reasonably concerned about censorship being introduced into the network right now.

Our take on this is simple: the MEV market is just beginning to develop, and the dominance that Flashbots has currently will likely not persist. Competitors will have room to enter the space via unique product offerings that capitalize on different MEV segments. A hopeful contributor to eliminating this centralization is Flashbots deciding to continue to open source the parts of their

tech stack that they can, submit blocks to other relays, and help clear some of the confusion regarding which relays are doing what:

Twitter

This paired with continued development of competitors in the space will undoubtedly contribute to less dominance. For example, let's look at what Manifold is doing with SushiGuard/OpenMEV.

Manifold offers the products of OpenMEV and SecureRPC, which is the Manifold relay. SushiSwap is going to be the first partner of Manifold's OpenMEV strategy by creating "SushiGuard", which helps:

- Protect users from MEV when making transactions
- Improve gas efficiency of transactions

Profits from this will be split 50/50 with Sushi and Manifold, and as we know Manifold's staking plans are to give 50/50 split with token holders and validators. Revenue projections for this implementation are pretty useless at this point, but it can go without saying that $35M of daily

volume on Sushi for the Ethereum network could lead to some good numbers going to $SUSHI and $FOLD stakers.

In addition to this, Manifold is planning on releasing their staking v2 contract soon, which will highlight the details of the (planned) 80/20 split between $FOLD and $ETH when staking, which will help bootstrap some token liquidity for $FOLD.

$FOLD provides one of the only *liquid* ways to participate in the potential upside of MEV in the future of Ethereum. While other protocols may arise or decide to launch a token, $FOLD undoubtedly has some first mover advantage for this specific use.

On the opposite side of this, founder Sam Bacha has been known to be dismissive of opposing ideas or frameworks. Which, historically, has not been beneficial for projects. Still, though, he talks very highly of the Manifold team and shows confidence that the middleware solution *will* be a dominant player in the MEV landscape.

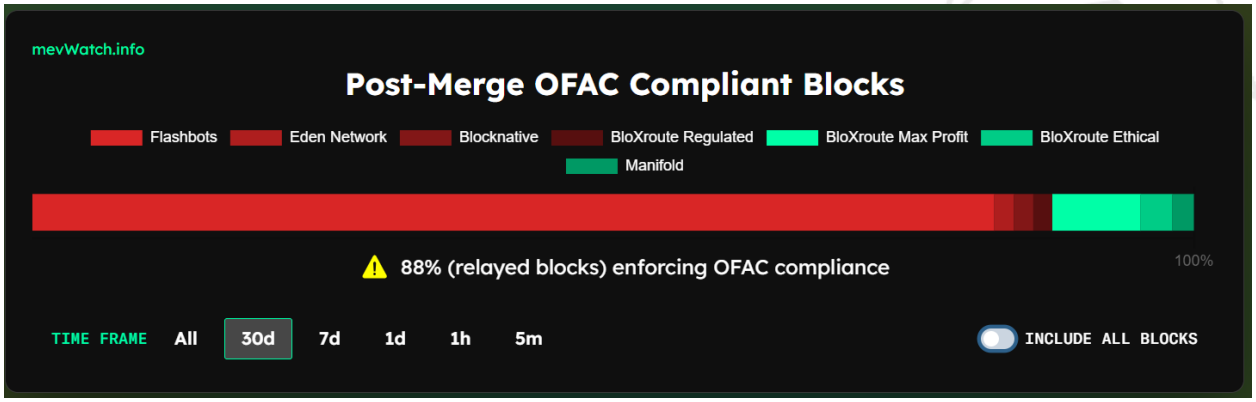
Here are some useful resources for the future of Manifold if you are interested in learning more:
Manifold Roadmap
Community call questions
Full Q&A breakdown

MEV Data
The data out there is anything but fact. Even though there are many resources out there that show MEV statistics, they should be taken with a grain of salt due to complications in actually *collecting* that data. Because of this, data can vary from one source to another, or not paint the whole picture.

For example, different builders will handle the transaction fee recipient address differently. The more “standard” method for the MEV-Boost relay is to have the transaction fee go to the builder address, with the final transaction in the block then forwarding that to the validator. However, other builders may handle this differently.

Regardless, MEV implementation is obviously very new. We are working on aggregating our own data for this right now, and will hopefully have an update in the near future with our own database. But for now, we will have to rely on what is out there. We can conclude some vague statistics.

- Flashbots has some arbitrary level of relayer dominance
- Most builders that use the Flashbots relay censor transactions (OFAC TC)



[MEVwatch.info]

Above shows that of the blocks that utilize a builder-relay, 88% are OFAC compliant with nearly 80% coming from Flashbots. Despite OFAC compliance being used synonymously with censorship on CT, it does not necessarily mean that. Some blocks may not include any OFAC (tornado cash) transactions, which would get flagged in the above data aggregator but not necessarily mean that these transactions are being excluded from Etheruem.

What the Future Could Look Like

Opportunities with MEV are vast, and frankly unpredictable. We are nowhere near experts on this subject matter and things are changing literally every single day. But, as always, it is worth an effort to try our hand at evaluating where things can be headed in the future for this unique crypto market.

Blockspace Markets

One of our first narrative pieces we wrote was about derivative markets in crypto, and in this we talked about what the team at Alkimiya is building with the blockspace commodity market.

This ties into MEV because MEV and block fees+tips are highly volatile. The futures market originally made sense for anyone who wanted to hedge out the risks that gas prices would rise at a given period, which would help smooth expenses and miners who want to smooth revenues.

But now we can apply these futures markets to the benefit of validators who want to sell future blockspace to builders who want to make a profitable MEV block. Sam with Manifold has already shown a clear interest in this and other future profitable concepts that can arise from developed and more sophisticated markets.

In the same vein as this, we liked Smol Thot's idea that there would be the development of a new DeFi primitive with some sort of prediction market/token based on future MEV yield. This would essentially be an extension of the blockspace market for builders and validators, but potentially open it to more speculative actors as well.

Privatized/Exclusive Order Flow

This section of MEV makes **a lot** of sense when thinking about the supply chain of MEV, which ultimately rests on the user's transaction intent. Whether routed through Infura's RPC into the public mempool or into Flashbots's private Protect RPC, a user wanting to swap one asset into another can open up the floodgates for MEV to be possible.

Which leads us to the thought: why would Uniswap not create their own MEV searcher/builder exclusively from the transactions that are sent to it?

If you send a trade to a Uniswap liquidity pool contract that has extractable value, that value *will be extracted.* No matter what. That is one of the largest considerations with MEV–someone stands to make a profit, and because participants of Ethereum are economically incentivized, someone will extract that profit.

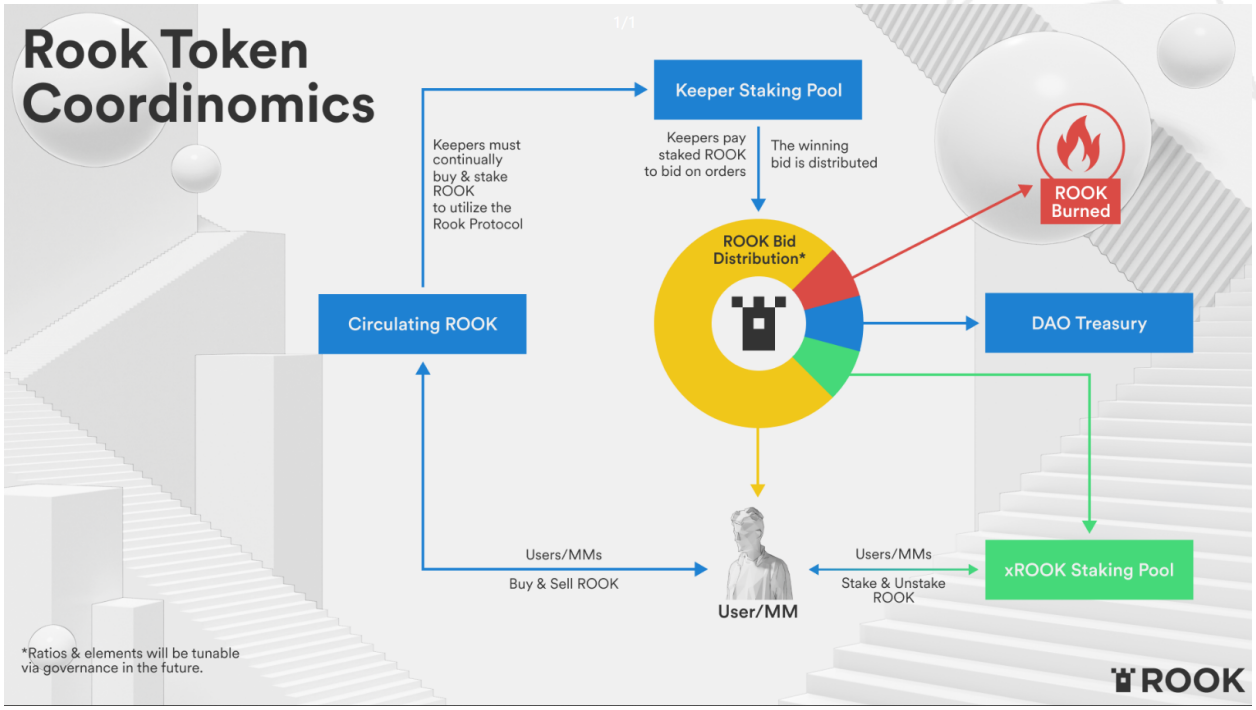
So back to the exclusive order flow (EOF). If Uni decided to open up their own private RPC, then they could do the following:

a) Spin up their own MEV bot
b) Build profitable blocks and submit as a bundle to ensure they get included
c) Distribute profits back to Uni stakers

Part C absolutely has some regulatory constraints and probably won't happen for the foreseeable future, but the point is the opportunities that are out there

In addition to Manifold utilizing SushiGuard to build profitable MEV blocks and distribute profits back to $FOLD stakers, the team at Rook has shown the potential of how EOF could work in practice.

With assigned "keepers" who are essentially searchers using proprietary methods to build profitable bundles, users can route their transactions through the Rook private RPC to get MEV protection AND a return of the profits from that MEV.

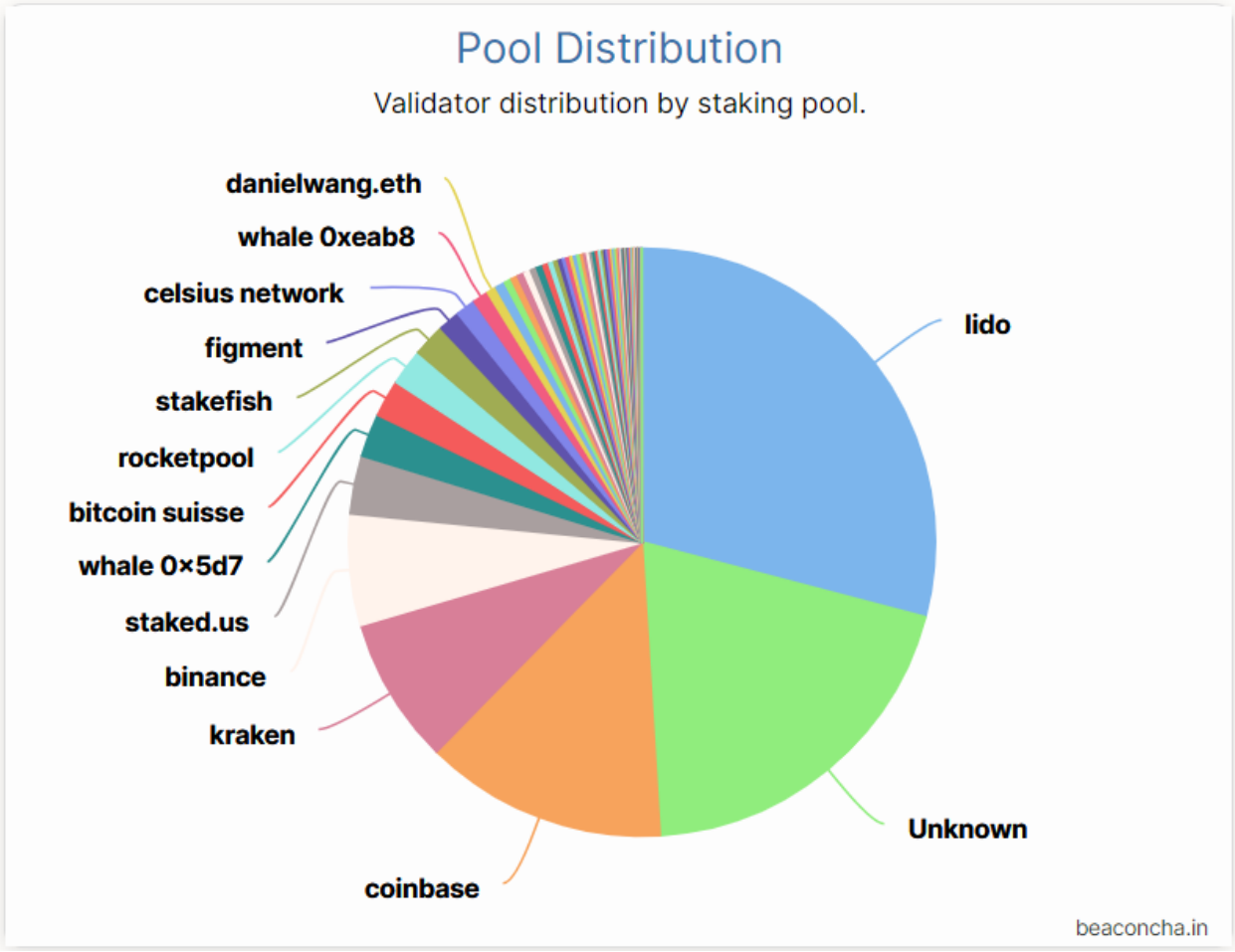


They have a similar staking method where $xROOK holders get to participate in the upside of the MEV profits earned by the keepers, with a large majority going back to the user. This is a simple implementation of MEV that shows how powerful and beneficial this can be for the end user.

Some Problems We're Thinking About

We've addressed the good and the bad of MEV. Solutions, drawbacks, future plans and ideas for the space. All of this to say that, despite the velocity of developments within the crypto industry in general, MEV will not be solved within a short time frame. There are issues today that will persist, and new ones that will come up.

Something that is inherently dangerous for the Ethereum network as a whole is the level of validator centralization. As the largest validators will have more opportunities to propose a block (block proposal weighted by $ETH stake), they will have more chances to *earn MEV profits.* More MEV profits, presumably being cycled back in to increase their $ETH stake, will give them even **more opportunities to earn on MEV** profitable blocks. This road leads to a high concentration of validators in the network, probably worse than we already see today.



[Beaconcha.in]

One of the biggest things that the Ethereum community wants to achieve is the ability to have a satisfactory level of decentralization. This comes with the ability to have low solo-staking nodes securing the chain.

With MEV, one of the constraints on solving some of the pressing issues (censorship resistance, too much EOF, fragmented builder industry) is ensuring that solo stakers *are indeed exposed* to the MEV markets, and can be just as well off financially as they would be with staking pools.

Data Issues

We've discussed a little how data is somewhat unreliable in this market and is essentially too nascent to draw any conclusions. To provide some color to that statement, we want to explain *why* this data is mixed in its reliability.

In a lot of Sam's seeming tirades on Twitter (or at least clear umbridging with certain takes and data presented in those takes), he has claimed data to be outright wrong. Here is what seems to be the issue:

When presenting a block via a relayer, a builder can have different ways to route the MEV revenue and gas tip (priority fee) to the validator and themselves. A lot of these data aggregators use the assumption of the Flashbots revenue structure, which is:

Fee Recipient Net Revenue = Priority Fees + MEV Revenue - Proposer Payment - Proposer Payment Gas Fee

In this structure, the payment comes as the last transaction in the block. But different builders may have different profit structures, so these dashboards and data centers don't account for all the earnings of a specific builder.

Conclusion

At the risk of sounding like a broken record, we do want to emphasize that MEV has a long way to go. The giga brains working on solving some of the pressing issues and creating the robust financial system that surround MEV are going to be facing problems they cannot predict or anticipate. As such, the market is heading in a direction that no one really knows.

But, it is clear that there is both demand and a requirement to work around some of the negative externalities present in MEV and ensure that users do not receive the short end of the stick.

Will this be a race to the bottom, where builders have to take less and less of a cut to be competitive? Or will the proprietary nature of block building/searching be enough to allow competitors to continue to take a large share of MEV profits?

There are a lot of unanswered questions, but we know that things change every single day, so it can only benefit us by staying on top of things.

Random Important Links:

https://inflectionvc.notion.site/Ethereum-post-merge-data-tools-66a29627f39d474ebc8c55b91f0e1785
https://0xfoobar.substack.com/p/ethereum-proof-of-stake
https://ethernodes.org/networkType/Hosting
https://docs.flashbots.net/new-to-mev