TLDR

- We studied the Ethereum block builder ecosystem and found a correlation between builder market share and searcher flow. Out of the 38 active builders identified, the top four had a combined 87% market share due to a significantly higher number of searchers submitting to them relative to other builders.
- We normalise builder market share to account for variances in searcher flow and find that this closes the gap between the top four builders and some smaller builders, suggesting they could compete with larger builders if they had similar flow.
- New builders face a 'chicken-and-egg' problem, relying on block subsidisation to gain market share and struggle with outreach and trust establishment. Searcher-builder entities have an advantage, being able to push searcher profits downstream to builders, but potential conflicts of interest may arise.
- We study searcher submission preferences and find that 88.5% of all searchers send to at least four or more different builders. We also find that 100% of the top twenty searchers send to at least five or more different builders.

Introduction

In this article we explore the dynamics of the Ethereum builder ecosystem, specifically focusing on the relationships between searcher flow and builder market share. Firstly, we categorise the current roster of builders into distinct groups based on their respective market share. We then examine the relationship between searcher acquisition and block construction, highlighting the effect of searcher count on market share. Following this, we introduce a series of normalised metrics to effectively rank builders and identify outliers. We conclude with discussions around searcher submission preferences and strategies, and discuss their impact on overall network decentralisation.

Data Collection

Searcher Database

Our analysis relies on an in-house database of known searcher contract addresses, which currently includes 612 unique contracts. Please note that not all of these contracts are active; some have been only used for a brief period due to the fast-paced and ever-evolving nature of searching.

This database uses the following three data sources:

Block Database

On each newly mined block, we fetch a variety of data points used in our analysis:

It is worth noting that throughout this analysis, when identifying which builders searchers submit bundles to, we can only consider mined blocks. As such, if a searcher submits bundles to a builder, but their bundles are never included in any of the builders' blocks, our analysis assumes that this specific builder never received flow from said searcher. Consequently, it is entirely plausible that builders might be receiving a higher searcher flow than our analysis indicates. However, based on our experience as block builders, it is quite rare for a competitive searcher to send bundles without ever landing in a block over a reasonable timeframe. That being said, we cannot guarantee that this applies to all builders.

Collection Summary

For the purpose of this analysis, we used data collected between 01/04/2023 and 17/05/2023. The dataset consists of 296,603 mev-boost blocks (~87.6% of all blocks). We identified 349 searchers from our searcher database that appeared in these blocks and 38 builders responsible for building them.

Accurate data on searcher occurrence could only be collected when the full mempool streaming service and database were operational. We flagged and excluded any block collected when the mempool service was unavailable to eliminate false positives. This precautionary measure prevents leaked re-org or missed slot bundles from appearing as private bundle flow sent to a builder.

We implemented a cut-off criteria, excluding searchers that landed fewer than 10 transactions over the period. This step was taken to curtail the false positive count of unique searcher-builder relationships. As a result, our total number of unique searcher contracts was reduced from 349 to 218.

Builder Ranking

Given the large number of builders identified, we have categorised them into groups for ease of reference in subsequent sections of this article. These categories are shown in Table 1 below. We exclude any builder responsible for constructing

less than 0.01% of blocks during the data collection period. The remaining builders are divided into three categories based on the overall percentage of blocks built: Top-4, Midfield and Long-tail.

Searcher Acquisition

This section looks at the cumulative searcher count and how it changes over time for each builder. We measure this by plotting the total number of searchers landing in each builders' blocks over the data collection period for all three builder categories.

Figure 1 indicates that the Top-4 builders are attracting new searchers at a similar rate. Despite being newer, rsync-builder is growing at a comparable pace. This suggests that new searchers are likely sending their transactions to all four of these builders. It is worth noting that new refers to any searcher that landed in a block during the data collection period. There may also be cases where some searchers were already sending flow to a builder but did not land frequently or at all during the collection period.

Looking at the Midfield category shown in Figure 2, Blocknative and eth-builder show strong and consistent Searcher growth. BuildAI also demonstrates significant growth. In contrast, Eden Network and BloXroutes' Searcher acquisition seems to have plateaued when compared with Blocknative, even though all three are infrastructure providers and Builders.

Manta-builder had no public RPC endpoint at the time of writing and built their last block on 12/05/2023. Similarly, finest artisanal blocks have not produced blocks since 08/05/2023.

Titan's public RPC went live on 17/04/2023 and has seen a steady increase in new searchers since.

Finally, we look at the Long-tail builders in Figure 3. The uptake of new searchers in this category is generally lower than the Midfield builders. Gambit Labs and nfactorial gained searchers at a commendable rate relative to other low market-share builders. A more notable exception is f1b, a new builder that launched on 11/05/2023 and has managed to acquire searchers at a rapid rate. Currently, f1b appears to be significantly subsidising its blocks in an effort to expand its market share.

A block subsidisation strategy involves paying the proposer a value greater than the total fees generated from all transactions in a block, effectively paying to build the block. This is also a common tactic among some larger builders, who use profits from high-value blocks to subsidise lower-value blocks in order to improve their market share. The effectiveness of this strategy indicates that subsidisation can be a worthwhile approach to attract searcher flow and consequently bootstrap market share.

Blocks Built vs Searcher Count

Figure 4 clearly highlights the distinction between the Top-4 builders and those in the Midfield category. The data generally aligns well with an exponential best-fit line, suggesting that as the count of searchers increases (and consequently, the bundle count), the permutations for bundle ordering increase exponentially, which may in turn allow more value to be extracted in each block. It is worth noting that this exponential relationship is simply an empirical observation, and worth pointing out that builder market share is also dependent on many other factors.

Interestingly, despite having a higher searcher count, the Flashbots builder does not seem to build as many blocks as the other builders within the Top-4. This underperformance could be due to different sorting strategies, less performant infrastructure or that the builder does not subsidise any blocks.

One notable overachiever in the Midfield category is bloXroute, whose performance deviates significantly above the line of best-fit. BloXroute operate their own relays, which potentially gives them a latency advantage when submitting blocks from their own builder. They also offer a variety of infrastructure products and services, which may earn them access to exclusive order flow (e.g. their Backrunme service).

Normalisation of Blocks Built

In this section, we attempt to normalise the percentage of blocks built to determine builder performance when accounting for the high variability in searcher count between builders. The reasoning behind this is that not all searchers are created equal; some searchers will have a higher impact on builder market share than others.

Manta-builder was excluded from the following analysis as an outlier due to its singular active searcher, which caused a significant spike in the normalised values. Manta-builder also does not have a public endpoint (i.e., it does not accept external flow) and operates an internal searcher. They have also ceased building blocks as of 12/05/2023.

Searcher Coverage

Firstly, we define Searcher Coverage SCBiSC_{B_i}SCBi for the i-th builder BiB_iBi as the percentage of searchers a builder receives bundles from:

Here, SBiS_{B_i}SBi is the number of searchers submitting to that builder i and STS_TST is the total number of searchers.

We then normalise the blocks built by Searcher Coverage to get the Blocks Built (Normalised) bBi,Nb_{B_i, N}bBi,N for the i-th builder:

where bBib_{B_i}bBi is the number of blocks built by the i-th builder and bTb_TbT is the total number of blocks built by all builders. Given that SCBi<1SC_{B_i} < 1SCBi<1 we scale Block Built (Normalised) by a constant R to ensure the sum of Blocks Built (Normalised) equals the sum of Blocks Built (%) allowing both values to be compared relative to each other.

When normalising for searcher coverage as shown in Figure 5, the differences among the Midfield builders become much more pronounced. The gap between the Top-4 builders and Midfield builders such as Titan and bloXroute becomes significantly smaller when normalised. This suggests that if these Midfield builders were to receive more flow, they could potentially gain significant market share.

Frequency Normalised Searcher Coverage

Next, we attempt to normalise the Searcher Coverage by considering the Searcher Inclusion Frequency. To do this we define a Frequency Weight WF,SjW_{F,S_j}WF,Sj for each searcher j:

where bSjb_{S_j}bSj is the number of blocks containing the j-th searcher, and the denominator is the sum of the number of blocks containing each searcher over all searchers from j=1 to STS_TST.

The top 10 searchers by frequency weight are shown in Table 2. It is clear from this table that one contract in particular (jaredfromsubway.eth) was responsible for over 30% of all MEV transactions over the period analysed.

We then define Searcher Coverage (Frequency Normalised) SCNF,BiSC_{{N_F},{B_i}}SCNF,Bi for each builder i, as:

which is calculated as the sum of the frequency weights for all searchers j that submit to the i-th builder.

The results in Figure 6 show a similar pattern to the previous analysis in Figure 5, but the relative differences between blocks built and the normalised values for some builders are less dramatic. This might suggest that although the Midfield builders have a lower searcher count, the searchers that do submit to them are high-impact in terms of block frequency.

Payment Normalised Searcher Coverage

Finally, we attempt to normalise Searcher Coverage by builder payment. Here we define a Payment Weight WP,SjW_{P,S_j}WP,Sj for each searcher j:

where FTSjF_{T_{S_j}}FTSj is the total internal transfers for the j-th searcher, FPSjF_{P_{S_j}}FPSj is the total priority fees paid by the j-th searcher, and the denominator is the sum of total priority and internal transfers over all searchers from j=1 to ST.S_T.ST.

The top 10 searchers by payment weight are shown in Table 3. Here jaredfromsubway.eth stands out even more, being responsible for ~45% of builder payments over the period. The Searchers highlighted in light green did not appear in the frequency-weighted list discussed previously (Table 2). Consequently, these searchers generated more builder value despite trading less frequently.

We then define Searcher Coverage (Payment Normalised) SCNP,BiSC_{{N_P},{B_i}}SCNP,Bi for each builder i, as:

which is calculated as the sum of the payment weights for all searchers j that submit to the i-th builder.

The results from this payment-weighted analysis align closely with those of the frequency-weighted approach. A notable exception is Finest Artisanal Blocks, which shows a larger spike in this latest analysis (Figure 7). This discrepancy appears to stem from the fact that Finest Artisanal Blocks did not receive searcher flow from jaredfromsubway.eth, while all other Midfield builders did. Jaredfromsubway.eth also had a higher payment weight than frequency weight, further supporting this hypothesis. This is further evidenced in Table 4 below, with Finest Artisinal Blocks having a significantly lower frequency and payment normalised Searcher Coverage relative to all other builders.

Searcher Submission Preferences

Figure 8 shows the distribution of the number of builders that each searcher sent bundles to. As shown in Figure 9, 88.5% of searchers send to four or more builders. However, there are a few searchers (5.5%) that only send to one builder, which suggests searchers that are run by the builders themselves.

As for builders that are known to operate an internal searcher (Searcher-Builders), we can mention a few that have openly disclosed their Searcher-Builder operation, such as eth-builder, BuildAI, manifold, and boba-builder. There are other builders that are less public about this relationship, but we leave identifying those as an exercise for the reader…

Figure 10 shows the same information as in Figure 8, filtered to only include the top 20 searchers as ranked by payment

weight. This suggests that these searchers are widely distributing their transactions across available builders, with 100% of searchers submitting to more than four builders as shown in Figure 11.

While the Top-4 builders dominate the market, not submitting to Midfield builders means missing out on 12.3% of all mev-boost blocks. Expanding on this, it is possible for a searcher to cover ~94% of blocks by submitting to just seven builders: the Top-4, BloXroute, Blocknative and Titan. This highlights the potential value of engaging with a diverse set of builders, rather than just sending bundles to the most prominent ones.

On the other hand, when evaluating a "shotgun" approach - that is, submitting to all known builders - it is also important to consider the balance between risk and reward. While this strategy maximises the chance of a searcher's transactions being included in a block, it also exposes the searcher to more risks. These include the unbundling of searcher bundles due to bugs in builder logic, possible information leakage for long-tail strategies, and tactical gas out-bidding by Searcher-Builder entities going after the same opportunities.

Conclusion

This analysis highlights the vital role that searcher flow plays in the success of Ethereum block builders and the decentralisation of the builder ecosystem. Here are the main insights drawn from our research:

Finally, to provide the reader with a sense of the substantial presence of private flow within the builder ecosystem, consider the following statistic: when analysing all transaction hashes in mined blocks, Titan receives all transactions that are eventually included in a block in only ~2.7% of cases. This percentage includes our mempool flow, bundle flow, and private transaction flow. Unfortunately, we cannot comment on what this statistic looks like for other builders, as we do not have access to their full internal flow, and can only analyse transactions that land on-chain in their blocks.

Although the present environment poses challenges for emerging block builders, it simultaneously unveils opportunities for innovation in building strategies. The participation of new builders is essential in promoting a more diverse builder ecosystem, which can disrupt the dominance of the Top-4 builders and facilitate more decentralisation.