

Data Availability Charges and Rollup Fee Expenditure in Ethereum

AUTHORS: David Kermaani

SUMMARY: This paper examines the impact of Ethereum's Dencun upgrade on data availability charges for rollups, focusing on Arbitrum, Base, Optimism, and zkSync Era. The study compares the observed blob burn against a calldata benchmark during a specific period, identifying significant differences in expenditure and transaction composition.

METHOD: The research uses a comparative analysis of fee expenditure before and after the Dencun upgrade. It employs a historical calldata gas schedule to benchmark blob burn and decodes pre-adoption payload gas to identify post-adoption expenditure components. The study also accounts for the fixed-composition and pooled intensity of transactions.

KEY FINDINGS:

- Blob burn was 97.63% below a calldata benchmark for the observed period.
- Nine congestion days accounted for 99.9986% of the observed burn.
- Decoded payload calldata accounted for 83.78% of Layer 1 expenditure.
- Excluding zkSync, pooled bytes per Layer 2 transaction fell 10.81%, and the fixed-composition index fell 6.91%.
- Including zkSync under a minimum-packing assumption, the pooled increase ranged from 3.71% to 9.48%, and the fixed-composition decline ranged from 6.29% to 18.56%.

IMPLICATION FOR TRADING: Understanding the specific fee structures and expenditure patterns of rollups can help traders and investors anticipate and manage costs associated with deploying and operating on Ethereum. This detailed analysis provides insights into the economic incidence of the Dencun upgrade, enabling more informed decision-making regarding the use of different rollups.

Data Availability Charges and Rollup Fee Expenditure in Ethereum

Working Paper Version 6.1 | September 14, 2026

David Kermaani | david@davidkermaani.com

Abstract

Ethereum's Dencun upgrade introduced a separate fee market for rollup data availability. For Arbitrum, Base, Optimism, and zkSync Era, observed blob burn was 97.63% below a same-inclusion-block calldata benchmark on the recorded physical representation during March 15-April 12, 2024. Nine congestion days account for 99.9986% of that burn; outside them, burn was approximately 5.11×10^{-7} of the benchmark. Decoded payload calldata accounted for 83.78% of pre-period Layer 1 expenditure. Receipts independently reproduce sampled blob burn and separate post-period execution burn, tips, and an aggregate residual. Quantity comparisons require a narrower definition: zkSync's dense encoded blobs reveal provisioned capacity, not unpadded pubdata. Excluding zkSync, pooled bytes per Layer 2 transaction fell 10.81% and the fixed-composition index fell 6.91%. Including zkSync under a documented minimum-packing assumption gives a pooled increase between 3.71% and 9.48% and a fixed-composition decline between 6.29% and 18.56%, conditional on the measured posting universe. The seven-day direction is not identified for the four-chain cohort. These results distinguish charges on specified representations from workload and composition. They do not identify causal savings or compression improvements.

Keywords: Ethereum; rollups; data availability; EIP-4844; fee expenditure; measurement

JEL classifications: L86, C81

1 Introduction

The cost of posting rollup data to Ethereum changed sharply after Dencun. Interpreting that change requires distinguishing the charge on an observed payload from changes in total expenditure, transaction composition, and the equilibrium prices of Ethereum resources. A rollup can publish more data and support more transactions while paying less to Layer 1 (L1). Neither transaction growth nor a lower observed charge alone identifies the economic incidence of an upgrade.

This paper studies Arbitrum, Base, Optimism, and zkSync Era around their initial blob adoption. The principal comparison uses February 13 to March 12 and March 15 to April 12, 2024, with 29 complete UTC days on each side. March 13 contains Dencun's activation and March 14 is the first positive blob day for each chain in the daily panel. Excluding both dates avoids mixing partial adoption days with complete days, although it does not prove that every subsequent payload used blobs.

The first measurement applies the historical calldata gas schedule to the physical-representation fields recorded for each sampled blob transaction, at that transaction's actual execution base fee. The resulting benchmark is 11,170.14 ETH against 264.89 ETH of recorded blob burn. This is a comparison between monetary charges on specified representations and dates. It excludes the execution overhead of constructing a hypothetical calldata posting transaction and does not estimate an alternative equilibrium.

The second measurement joins decoded pre-adoption payload gas to posting receipts and aggregate expenditure. It identifies 9,118.49 ETH of payload base-fee burn within 10,884.26 ETH of aggregate expenditure. For the post period, receipts for all 41,068 sampled blob transactions independently verify blob burn and separate execution burn and tips. Both periods reconcile the selected posting receipts against aggregate expenditure, with the remaining difference reported explicitly.

The third measurement separates pooled intensity from fixed-composition intensity while accounting for a chain-specific measurement limit. For Arbitrum, Base, and Optimism, the approximate packing-adjusted pooled ratio falls 10.81% and the fixed-composition ratio falls 6.91%. For zkSync, the post variable is provisioned capacity in logical units. Under minimum packing, the four-chain pooled change lies between +3.71% and +9.48%, while the fixed-composition change lies between -18.56% and -6.29%. These intervals concern the sampled blob payloads and depend on the documented packing rule; additional posting routes remain a separate coverage question. No four-chain seven-day sign reversal is established.

The contribution is a reproducible accounting of specified payload charges, pre- and post-period receipt components, and chain composition within one cohort. It connects known fee-market developments to the quantities and expenditure components measured here. The replication package supplies transaction inputs, new remote procedure call (RPC) receipt, block-price, and nonce responses, daily series, calculation code, and supplementary table inputs. Raw-body decoding and an exhaustive posting-route census remain outside the validation completed in this paper.

2 Institutions and prior evidence

Ethereum activated Dencun on March 13, 2024 at 13:55 UTC (Ethereum Foundation 2024). EIP-2028 specifies the execution-gas charge relevant here: four gas for a zero calldata byte and sixteen for a nonzero byte. EIP-4844 introduced a separate data availability (DA) resource. At launch, each blob contained 131,072 bytes and incurred 131,072 blob gas, with a target of three blobs per block, a maximum of six, and a minimum blob base fee of one wei per blob gas. The controller responds to accumulated excess usage rather than clearing a spot auction independently in each block (Buterin et al. 2022).

Blob transactions also consume execution gas and can pay execution priority fees. Blob base-fee burn, execution base-fee burn, and tips must therefore be distinguished. L1 expenditure in this paper includes payments attributed to rollups in the aggregate source; it is not economic rent, validator revenue, L2 user fees,

or sequencer profit. The resources also provide different retention arrangements. Equal byte counts or equal numerical gas prices do not imply equal economic services or physical costs.

Heimbach and Millionis (2025) analyze the early blob market, including congestion, packing, and posting behavior. Huang et al. (2024, version 1) study builder and rollup strategies at large scale. Blocknative's contemporary analysis identifies the March 27 BlobScriptions shock and its subsequent decline; Ved and Andrade (2024) document blob adoption and fee-market developments. The present regime comparisons describe this known episode within the four-rollup sample. They do not establish a new causal explanation for aggregate blob demand.

Posting behavior and compression are also established research subjects. Mamageishvili and Felten (2023) analyze batch-posting strategy on the base layer; Crapis, Felten, and Mamageishvili (2023) model EIP-4844 economics and rollup choices. Palakkal, Gorzny, and Derka (2024) survey rollup compression. Relative to those contributions, this paper's tables link a charge adjusted for provider-reported trailing padding at the same inclusion block to decoded receipt expenditure and to both pooled and fixed-composition quantities. The linkage makes it possible to distinguish changes in posting charges from changes in cohort composition.

The OP Stack Delta upgrade became available on February 22, 2024, inside the pre-window. Its span-batch format can change data requirements. Protocol activation does not establish the first day an operator actually emitted span batches: the upgrade proposal explicitly requires batcher enablement. Consequently, restricting dates after Delta is a sensitivity check, not a verified control for compression policy (Test in Prod 2024; Sunnyside Labs 2024).

3 Measurement framework

3.1 Charges and fee components

For a posting receipt j , let g_j be gas used, p_j the block execution base fee, e_j the effective gas price, and c_j the gas attributed to decoded payload calldata. The pre-adoption components are $c_j p_j$, $(g_j - c_j)p_j$, and $g_j(e_j - p_j)$. Their sum is $g_j e_j$. The second component includes all remaining receipt gas; it is not a direct classification of individual EVM operations.

For the post sample, let U_j denote the sum of provider-reported physical-representation bytes, N_j their implied nonzero-byte count, H_j provisioned blob gas, and d_j the supplied blob fee. Define the benchmark and recorded blob burn by:

$$C_j = 4 U_j + 12 N_j; \quad D_C = \sum_j C_j p_j; \quad D_B = \sum_j d_j. \quad (1)$$

The implementation obtains C_j by subtracting four times the reported unused padding from Blobscan's full-representation calldata gas. Thus N_j can be recovered algebraically as $(C_j - 4 U_j)/12$, but this is an internal consistency check rather than an independent byte recount. The benchmark retains the physical representation's encoding overhead. It does not reconstruct the counterfactual compressed calldata stream that a rollup would actually choose.

Actual blob payments charge provisioned capacity while D_C excludes 175,497,018 reported padding bytes. Pricing that padding as zero bytes at each transaction's inclusion price adds 18.48 ETH, or 0.1654% of D_C . The corresponding full-capacity reduction is 97.6325%, compared with 97.6285% for physical usage. This adjustment concerns reported trailing zeros only. It does not measure padding embedded in zkSync's transformed representation (Section 3.3).

The recorded physical usage is 26,021,077,190 bytes, with an implied nonzero share of 99.3769% and an average charge of 15.9252 gas per byte. Treating every usage byte as zero gives a 2,805.75 ETH benchmark and a 90.56% reduction; treating every byte as nonzero gives 11,223.00 ETH and a 97.64% reduction. Reducing both usage and its gas proportionally by 20% gives a 97.04% reduction. These specified perturbations show numerical sensitivity; they do not independently validate usage lengths or chain framing.

A separate representation sensitivity replaces zkSync's encoded bytes with hypothetical logical pubdata lengths while keeping other chains' benchmark components unchanged. Applying the full-window minimum fill fraction proportionally to zkSync's physical charge gives 430.57 ETH and a cohort reduction of 97.4925%; this is an aggregate-price heuristic. Pricing each posting's minimum logical length at its actual block price and sixteen gas per byte gives 395.77 ETH and 97.4843%. Allowing four to sixteen gas per logical byte and minimum length to capacity gives a conditional cohort reduction range of 97.4113% to 97.6223%. These are alternative-representation sensitivities, not a recovered calldata posting charge. They depend on the packing and coverage assumptions below; the physical-representation result remains 97.6285%.

3.2 An expenditure identity

Blob gas and execution gas are different accounting units. Rescaling one resource's numerical units while inversely rescaling its price leaves monetary payments unchanged. Economic resource-versus-price shares cannot be inferred from a log decomposition that changes under this transformation.

A useful accounting identity keeps the quantity terms in historical calldata-gas units. Let G_0 and D_0 be pre-period decoded payload gas and burn; let G_1 be post-period calldata-equivalent gas. Define $P_0 = D_0/G_0$ and $P_1 = D_1/G_1$. For any two windows:

$$D_B / D_0 = (G_1 / G_0) (P_1 / P_0) (D_B / D_C). \quad (2)$$

The terms describe measured calldata-equivalent quantity, a ratio of payload-weighted execution prices, and the observed same-payload charge ratio. P_0 and P_1 use different baskets and inclusion times. The final factor incorporates resource choice, representation, capacity utilization, and market conditions; it is not an isolated technological effect. Equal duration is needed only to interpret G_1/G_0 as a ratio of daily rates. Equation (2) is an accounting bridge rather than a causal attribution.

3.3 Intensity and composition

For each chain i and window w , let b_{iw} be measured payload bytes divided by L2 transactions, both summed over the window. Let s_{iw} be its share of cohort transactions. The pooled ratio compares $\sum_i s_{i1} b_{i1}$ with $\sum_i s_{i0} b_{i0}$. The fixed-composition Laspeyres ratio instead holds the weights at s_{i0} :

$$L = (\sum_i s_{i0} b_{i1}) / (\sum_i s_{i0} b_{i0}). \quad (3)$$

For the three optimistic rollups, the pooled series uses observed transaction shares and the fixed-composition series holds pre shares fixed. Post representation lengths are multiplied approximately by 127/128: Nitro packs 254 bits per field element, and the OP Stack packs 127 bytes per four field elements, with framing overhead. The archived March 2024 implementations document these conventions (Offchain Labs 2024; OP Labs 2024). This adjustment is not a full decoder; it does not remove all framing, trailing-zero ambiguity, or within-chain workload changes.

zkSync requires separate treatment. In the archived March 14, 2024 source, `KzgInfo::new pads pubdata to 126,976 bytes and constructs a polynomial, applying a fast Fourier transform and bit reversal; pubdata_to_blob_commitments` chunks the input into pieces of that size (Matter Labs 2024). The Ethereum blob is a transformed representation. Zero padding of polynomial coefficients need not appear as trailing zero bytes in that representation. In the selected sample, 10,682 postings contain 18,273 blobs whose reported usage totals 2,395,078,578 bytes, only 78 bytes below full capacity. The implied nonzero share is 99.5945%. Multiplying full physical capacity by 31/32 gives exact logical capacity, not unpadding logical usage; applying the factor to the provider-trimmed length instead omits the logical-capacity equivalent of those 78 physical bytes. The exact capacity is 18,273 times 126,976 bytes.

The pre-period extract records two batches in 11 of 25,840 main-file zkSync commits. This confirms that one batch per posting is not a universal property of the pre-period route; it does not establish the batch count of post-period blob postings.

Let $K = 126,976$ and m_j be the observed blob count for posting j . Conditional on each posting representing one nonempty pubdata batch and using the minimum number of blobs required by the documented chunking rule, logical length q_j satisfies:

$$K(m_j - 1) + 1 \leq q_j \leq K m_j.$$

The bounds sum over postings before division by window transaction counts. The sample has 3,091 one-blob and 7,591 two-blob postings. The source supports this minimal-packing interpretation; the archive does not independently authenticate every historical batch mapping or operator binary. The intervals are consequently conditional, not unconditional raw-byte measurements. If extra blobs were deliberately provisioned or a transaction combined independently padded batches, the stated lower bound would need revision. Capacity remains an upper bound on the logical content of the selected blobs. Without minimum packing, a zero logical-byte lower bound is conservative and the sign of the four-chain pooled change need not be positive.

The primary quantity comparison therefore uses Arbitrum, Base, and Optimism. Four-chain results are reported as intervals, holding the three-chain proxies fixed and varying zkSync within the conditional packing bounds. These are identification intervals, not sampling confidence intervals. All quantity results describe selected postings. Bytes are dated by L1 inclusion and transactions by L2 block date; unknown posting lag can move data across window edges, including across the March 13-14 exclusion. Its magnitude is not measured here, and short windows are particularly exposed.

4 Data and validation

4.1 Archived inputs and estimation coverage

The daily panel has 5,724 observations for six rollups from February 1, 2024 through September 11, 2026. The transaction-level post extraction contains 217,094 observations in its wider coverage; the pre-period receipt file contains 189,033. The principal four-chain windows use 41,068 blob-bearing postings after adoption and 133,235 decoded postings before it. Each principal period has 116 rollup-days. The panel has 126 missing expenditure observations: 120 for zkSync beginning July 29, 2025, outside all estimation windows, and six for Linea, which is outside the four-chain cohort. Post-period panel blob totals match the transaction extract exactly: 199,864 blobs, 264.8942316900262 ETH, and 26,196,574,208 blob gas. This establishes completeness relative to the provider attribution. The key-specific checks below provide additional, narrower evidence of on-chain coverage.

Table 1 Principal sample and observable coverage

Period	Inclusive UTC dates	Days	Posting records
Pre	Feb 13-Mar 12, 2024	29	133,235
Post	Mar 15-Apr 12, 2024	29	41,068

March 13-14 are excluded. Four chains in each period. Pre records are decoded postings; post records are blob-bearing postings. The universes are not asserted to cover identical classes of L1 operations.

The pre payload object is the recognized SequencerInbox data argument for Arbitrum, the authenticated batcher frame stream after its version byte for the OP Stack chains, and commit-batch pubdata for zkSync. A transition supplement repairs the zkSync validator path in March. OP frame streams include transport structure; zkSync pubdata includes different semantic objects from optimistic-rollup compressed transaction streams. Cross-chain intensity levels are therefore operational measurements, not comparable units of economic output.

4.2 What the checks establish

The offline audit verifies unique transaction hashes in each extraction, the provisioned-size and padding identities, nonnegative and integral implied nonzero-byte counts bounded by usage, valid pre-receipt gas relationships, and fee multiplication from integer fields. Calldata benchmark totals use arbitrary-precision

integer multiplication before conversion to ETH. This avoids signed 64-bit overflow in the legacy ETH_V2_calldata_counterfactual_daily.csv columns calldata_cf_burn_wei and actual_blob_fee_wei when a sum exceeds $2^{63} - 1$ wei, approximately 9.22 ETH. The transaction integer fields are used to regenerate daily totals. These checks reproduce the headline charge and pre-period payer fees from distributed inputs.

The original notebook copied blob fees from Blobscan's transaction field named blobGasBaseFee. The receipt check instead multiplies blobGasUsed by blobGasPrice from newly retrieved RPC receipts for every selected post transaction. Execution gas used and effective gas price come from those receipts; execution base fees are checked independently with eth_feeHistory for every block in the post window. Hashes, block numbers, transaction type, receipt status, provisioned blob gas, and integer fee products are checked before aggregation. All 41,068 receipts are successful type-3 transactions. Receipt blob fees and provisioned blob gas agree exactly with the archived values for every transaction. All archived execution prices also agree with the new block-price responses. The full-window base-fee series contains 206,308 blocks.

The byte field meaningful_payload_bytes remains the sum of Blobscan usageSize fields. The frozen materials do not establish a separately verified, chain-specific termination rule for each blob's usage. A raw-body audit must distinguish trailing zeros from meaningful zero-valued data and validate chain framing or length information before interpreting the trimmed string as the true logical payload.

The extraction selects blob-bearing transactions and contains no raw blob bodies. The completed receipt check establishes the sampled fee components, but it does not enumerate every additional posting or decode possible calldata carried alongside blobs. Post intensity uses approximate packing-adjusted representation lengths for the three optimistic rollups and logical-capacity intervals for zkSync. Additional calldata payload would increase the relevant numerator. Appendix A gives conditional outer bounds, allowing for low execution prices, zero bytes, gas refunds, and possible mixed-resource postings. The same-payload charge ratio describes the selected blob observations.

4.3 Activity and aggregate attribution

The reviewed growthepie source snapshot maps the standard transaction metric to filtered counts: gas_price greater than zero for standard EVM queries and gas_used greater than zero for Arbitrum. The three archived query/configuration files are byte-for-byte identical to repository commit 2bea111699a0cb38e17f1a0c5f4897ef976fbb9d, dated September 10, 2026. This pins the reviewed source; it does not prove which revision generated each historical observation. Raw L2-day count reconciliation and historical query provenance are not present. The archive also lacks a pinned address-and-operation definition of rent_paid_eth; expenditure reconciliation remains conditional on that metric's attribution. Stable measurement across windows is necessary for a comparable ratio; unchanged filters alone do not make workload composition or economic value constant.

Aggregate L1 expenditure is kept distinct from the narrower posting-receipt universe. A positive reconciliation residual is not labeled prove, execute, assertion, or output-proposal expenditure without a receipt and selector match. The statistical unit of the event comparison is the chain and calendar window; thousands of posting receipts do not supply thousands of independent upgrade assignments.

5 Results

5.1 Same-payload charges and price sensitivity

Recorded blob burn totals 264.89 ETH, compared with 11,170.14 ETH under the physical-usage calldata benchmark. The aggregate ratio is 0.0237145, a 97.63% reduction. Chain-level results are shown in Table 2. The chains share the same execution and blob markets over overlapping dates, and their recorded representations cost close to sixteen calldata gas per byte. Their similar reductions therefore reflect common prices and similar gas per byte; they do not constitute four independent tests of a mechanism.

Table 2 Recorded blob burn and calldata benchmarks

Chain	Blob burn ETH	Actual block price ETH	Pre scalar price ETH	Reduction
Arbitrum	65.29	3,089.76	5,172.03	97.89%
Base	111.88	4,457.06	7,908.90	97.49%
Optimism	67.91	2,586.87	4,369.44	97.37%
zkSync Era	19.82	1,036.45	1,772.22	98.09%
Total	264.89	11,170.14	19,222.59	97.63%

Entries are rounded independently from unrounded inputs. March 15-April 12. Both calldata columns price the same recorded physical-representation gas quantity. The scalar price is 46.3875 gwei. Reduction compares blob burn with the actual-block-price benchmark; it excludes hypothetical transaction overhead.

Execution prices in this benchmark are post-adoption prices. Moving data out of the execution market could itself change congestion, so they cannot be interpreted as unaffected no-Dencun prices. Repricing all measured post calldata-equivalent gas at the scalar pre-period payload-weighted price gives 19,222.59 ETH. This alternative fixes a price level while leaving observed quantities unchanged; neither benchmark bounds the missing equilibrium.

Table 2 prices recorded physical representations. The three-chain intensity series uses approximate packing adjustments; zkSync is represented by logical-capacity bounds. Scaling charges by these packing or capacity-conversion factors yields 11,058.58 ETH and a 97.60% reduction. This proportional sensitivity does not remove zkSync's hidden pubdata padding or recover logical-byte composition. Exact logical-byte repricing requires decoded bodies rather than scaling a gas total.

5.2 The initial price regime

The aggregate charge ratio is a calldata-benchmark-weighted average of daily ratios, not their unweighted mean or median. It is also not a permanent discount. The median daily ratio is 3.42×10^{-10} percent, while the ratio of aggregate charges is 2.371%. Table 3 makes the difference between ordinary days and the BlobScriptions episode explicit.

Table 3 Charges across observed price regimes

UTC period	Days	Blob ETH	Calldata ETH	Ratio %
Mar 15-26	12	9.33e-09	4,209.54	2.22e-10
Mar 27-Apr 4	9	264.891	4,070.40	6.508
Apr 5-12	8	3.63e-03	2,890.19	1.26e-04
Excluding Mar 27-Apr 4	20	3.63e-03	7,099.73	5.11e-05
Excluding five largest burn days	24	48.761	8,876.51	0.549
Full window	29	264.894	11,170.14	2.371

Ratio is 100 times summed blob burn divided by summed calldata charges. Full-window median daily ratio: 3.42×10^{-10} percent. The two exclusion rows overlap the three calendar subwindows and are sensitivity calculations, not additional observations.

March 27 to April 4 contains 99.9986% of sampled blob burn; within that interval the aggregate blob-to-calldata ratio is 6.51%. Outside it, the ratio is approximately 5.11×10^{-7} . Excluding the five highest-burn dates instead gives a 0.55% ratio. These exclusions answer different descriptive questions and are not estimates of a world without inscriptions. The episode dates follow contemporary reporting (Marshall 2024).

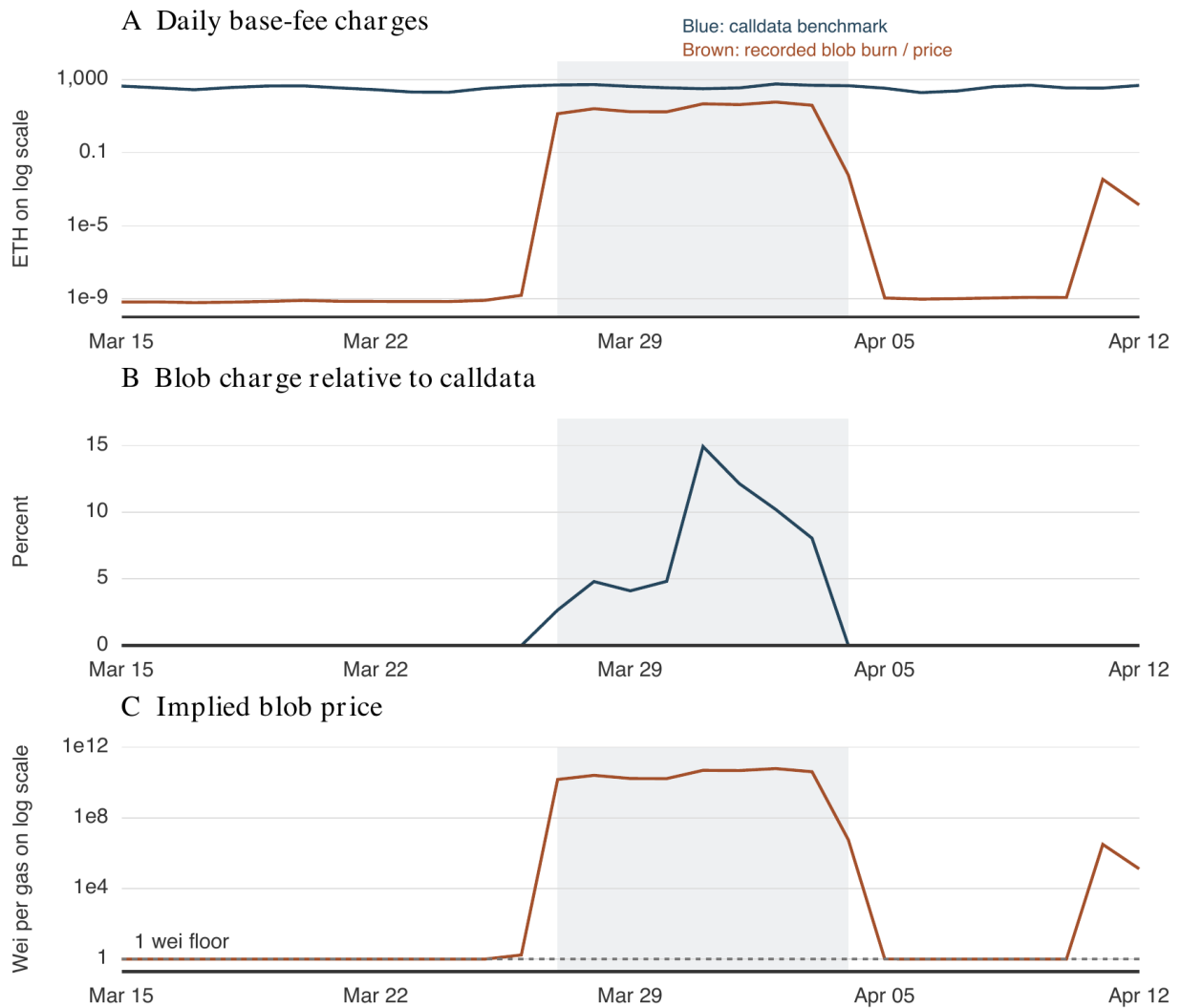


Figure 1 Daily charges, charge ratios, and implied blob prices. Shading marks March 27–April 4, the BlobScript interval. Panel C shows the one-wei floor explicitly; its price is inferred from recorded fees divided by provisioned gas. Source: frozen transaction records.

Daily aggregation can conceal individual posting choices. One Base observation on March 27 has only 125 reported usage bytes in a provisioned blob and a blob charge 53.68 times its physical-usage calldata benchmark. Thus the aggregate discount does not mean every observed posting was cheaper in blob form. Conversely, a daily ratio below one cannot establish that no rollup switched resources. The nonce check below excludes additional transactions from the three observed optimistic-rollup posting keys, while other keys and mixed calldata still require separate coverage checks.

The cohort's average demand is below the original global target, but this fact alone cannot establish that it never contributed to congestion. Other users, temporal clustering, and accumulated excess blob gas enter the controller. The event interpretation relies on the broader market literature; the present sample quantifies its own expenditure concentration.

5.3 Receipt components and expenditure reconciliation

Decoded payload burn is 83.78% of aggregate pre-period expenditure. The share is 91.95% for Arbitrum, Base, and Optimism together, compared with 59.75% for zkSync; Base alone is 97.72%. Table 4 reports the receipt components and unclassified residual separately.

Table 4 Pre-period receipt components and aggregate reconciliation

Chain	Payload burn	Other burn	Tips	Residual	Aggregate	Payload share
Arbitrum	3,012.7	506.6	3.7	17.6	3,540.7	85.09%
Base	1,689.6	35.3	1.3	2.8	1,729.1	97.72%
Optimism	2,764.9	31.5	51.6	2.9	2,850.9	96.98%
zkSync Era	1,651.2	261.6	219.9	630.8	2,763.6	59.75%
Total	9,118.5	835.1	276.5	654.1	10,884.3	83.78%

ETH over February 13-March 12. Residual is aggregate expenditure minus reconstructed posting payer fees. It is not identified proving or settlement expenditure. Payload share uses aggregate expenditure as denominator.

The total unclassified residual is 654.11 ETH, including 630.80 ETH for zkSync. Its architecture permits distinct commit, prove, and execute operations, but that does not identify the residual empirically. The reported payload fraction measures the decoded component relative to the aggregate series; it is not asserted to be an exhaustive DA share. No conclusion about the composition of omitted expenditure follows from its sign alone.

Arbitrum's other receipt burn merits a separate check. Of its 506.63 ETH, only 2.83 ETH is associated with whole-input calldata gas outside the decoded payload. The remaining 503.80 ETH is attributable to receipt gas beyond all input calldata gas. The payload covers 99.91% of whole-input calldata gas in this sample. The other burn primarily reflects gas beyond input calldata. The remaining amount comprises 40.24 ETH for the standard 21,000 gas per call and 463.56 ETH of additional net gas. That additional gas may include access-list and refund effects; exact opcode attribution requires traces.

New post-period receipts complete the corresponding selected-sample decomposition. Of 583.33 ETH in aggregate post expenditure, 264.89 ETH is blob burn, 119.08 ETH is sampled execution base-fee burn, and 7.96 ETH is sampled tips. The selected receipts sum to 391.93 ETH, leaving 191.40 ETH outside that sample. zkSync accounts for 178.25 ETH of the residual. Every chain-day residual is positive; the smallest is 0.0284 ETH.

Table 5 Post-period receipt components and reconciliation

Chain	Blob burn	Execution burn	Tips	Residual	Aggregate
Arbitrum	65.285	43.019	1.563	9.814	119.682
Base	111.877	7.582	0.612	1.683	121.755
Optimism	67.910	4.403	0.326	1.655	74.293
zkSync Era	19.822	64.074	5.454	178.253	267.603
Total	264.894	119.078	7.956	191.404	583.333

ETH over March 15-April 12. The complete selected blob sample is checked against RPC receipts. Residual equals aggregate expenditure minus the selected receipts including both fee markets. It is unclassified and conditional on aggregate attribution. Entries are rounded independently.

The archived nonce responses count all transactions sent by each observed key between block states immediately before and at the end of each window. Counts for Arbitrum, Base, and Optimism match the unique sampled hashes: 39,946/8,959, 36,434/13,804, and 30,282/7,623 pre/post respectively. This supports a census of those keys conditional on the archived sender attribution; it does not exclude other keys, internal posting routes, or calldata inside blob transactions. The nonce audit and adjacent boundary-block timestamps reproduce offline.

For zkSync, the old key sent 52,768 pre and 11,132 post transactions; 25,840 pre transactions and no post transactions from that key are in the sender-labelled payload/receipt sample. The new key sent 733 pre and

10,682 post transactions, with all 10,682 post receipts in the sample. The pre fee supplement has 733 rows, but its sender attribution is not independently proved by matching counts. These nonce differences identify candidate omissions; they do not establish that their fees equal the expenditure residual. That requires a transaction and fee census.

The residual in Table 5 covers expenditure outside the selected receipts only if the aggregate series completely and consistently attributes the relevant payer universe. Close aggregate reconciliation is supportive evidence, but does not by itself prove route completeness or exclude offsetting attribution errors. Calldata within sampled blob transactions is part of their execution component, not this residual.

Equation (2) yields a calldata-equivalent gas ratio of 2.1081, an execution-price ratio of 0.5811, and a same-payload charge ratio of 0.0237145. Their product is 0.02905, equal to post blob burn divided by pre decoded payload burn. The execution prices are 46.39 and 26.96 gwei under their respective payload weights. This exact identity describes how recorded spending relates to measured quantity and price aggregates without assigning economic shares to arbitrary gas units.

Table 6 Arithmetic contributions to the observed expenditure decline

Component	Pre ETH/day	Post ETH/day	Decline points
Measured payload burn	314.43	9.13	81.34
Aggregate minus measured payload burn	60.89	10.98	13.30

Contributions divide each component decline by total pre expenditure; they sum to 94.64 percentage points. The post remainder is further reconciled in Table 5. This is not causal attribution.

The observed daily expenditure decline divides arithmetically into 81.34 percentage points from the measured payload-burn difference and 13.30 points from aggregate expenditure minus measured payload burn. The second term falls from 60.89 to 10.98 ETH per day. Table 5 now separates its post value into sampled execution burn, sampled tips, and the aggregate residual. Execution gas can include calldata carried within blob-bearing transactions. These are payment components, not causal channels.

5.4 Pooled and fixed-composition intensity

The three-chain cohort excludes zkSync's capacity measure and renormalizes transaction shares within Arbitrum, Base, and Optimism. Over 29 days, pooled intensity falls 10.81% and the Laspeyres index falls 6.91%. This is a comparison of measured posting proxies, not an estimate of compression efficiency. Table 7 reports chain levels; zkSync is explicitly an interval.

Table 7 Principal-window chain intensity and zkSync bounds

Chain	Pre bytes/tx	Post bytes/tx	Change
Arbitrum	170.19	157.84	-7.26%
Base	186.66	171.75	-7.99%
Optimism	312.82	294.57	-5.83%
zkSync Era	72.16	28.76 to 69.22	-60.15 to -4.07%

zkSync post values range from the conditional minimum-packing length to provisioned logical capacity. Other post values are approximate packing-adjusted proxies. The intervals hold the selected posting universe fixed and are not sampling confidence intervals.

For all four chains, pre payload totals 12.425 billion bytes. Under minimum packing, the pooled change lies between +3.71% and +9.48%, and the Laspeyres change between -18.56% and -6.29%. The post endpoints combine observed optimistic-rollup proxies with zkSync's conditional logical-length bounds. Holding pre intensities fixed and changing transaction shares raises the pooled level by 17.67%; that composition calculation uses pre intensities only. It helps explain why the four-chain pooled interval is positive while

within-chain fixed-composition comparisons decline. The positive pooled sign is conditional on minimum packing and on the measured numerator definition.

Table 8 reports each comparison using its own pre transaction weights. In the three-chain cohort, both indices decline over seven, fourteen, and twenty-nine days. In the four-chain cohort, the seven-day Laspeyres interval is -16.35% to +2.21%; its direction is not identified. The corresponding zkSync change is -65.03% to +56.74%. Neither the seven-day reversal nor a zkSync increase is established.

Table 8 Cohort and window sensitivity

Window	Cohort	Pooled change	Laspeyres change
7 days	Three chains	-9.05%	-7.60%
7 days	Four chains	-2.86 to 9.51%	-16.35 to 2.21%
14 days	Three chains	-11.63%	-9.93%
14 days	Four chains	-3.33 to 4.99%	-17.83 to -4.69%
29 days	Three chains	-10.81%	-6.91%
29 days	Four chains	3.71 to 9.48%	-18.56 to -6.29%
After Delta date	Three chains	-8.83%	-4.67%
After Delta date	Four chains	7.52 to 13.50%	-15.26 to -2.59%

Seven-day pre/post: March 6-12/March 15-21. Fourteen-day: February 28-March 12/March 15-28. Main: February 13-March 12/March 15-April 12. After Delta date: February 23-March 12/March 15-April 12. Each comparison uses its own pre weights. Four-chain intervals assume one nonempty batch per posting and minimal packing.

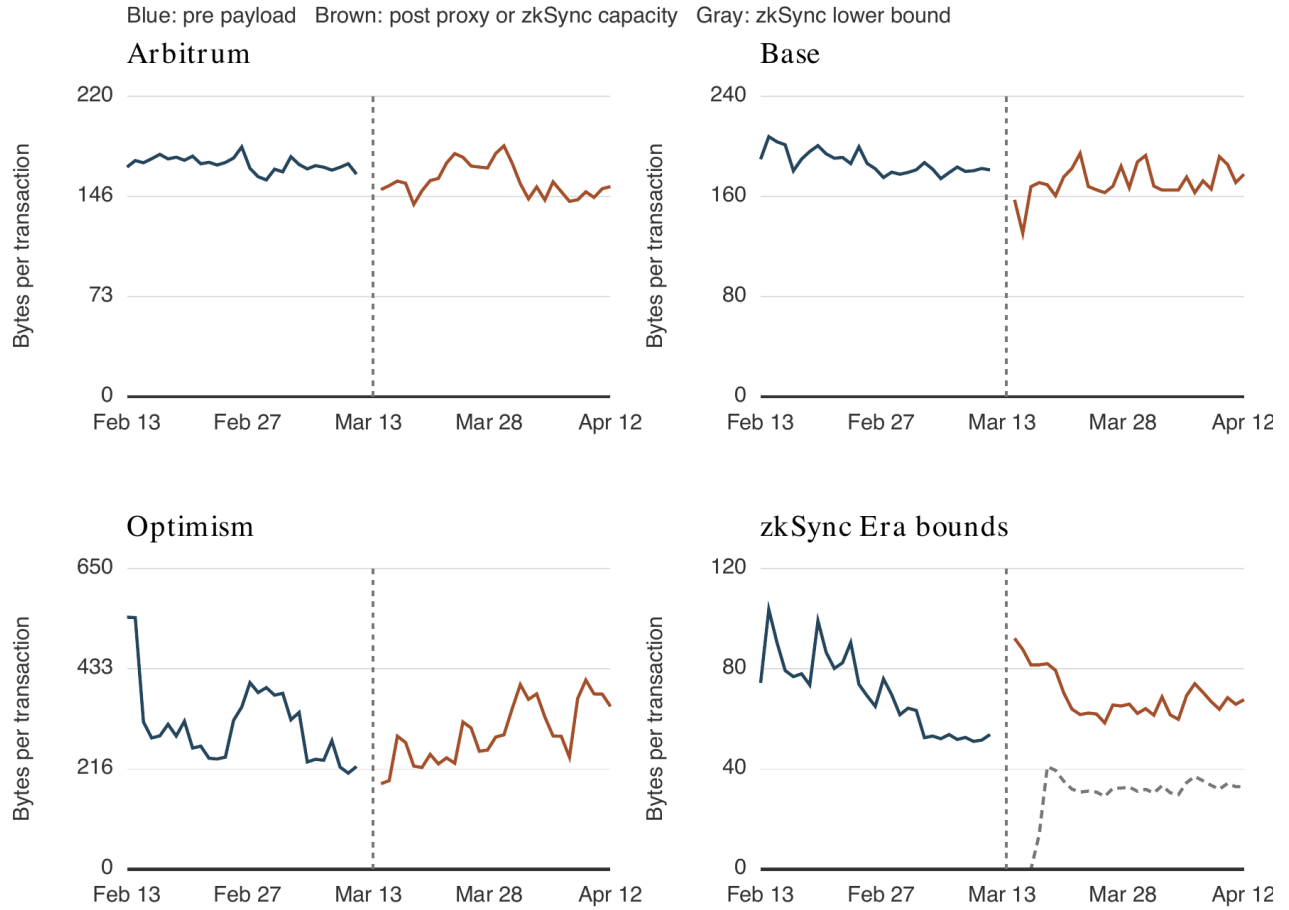


Figure 2 Daily payload proxies and conditional zkSync bounds. The dotted line marks March 14, the first-positive blob day in the daily panel; it is not an exact first-transaction timestamp. March 13-14 are excluded. Pre bytes are decoded payloads; optimistic-rollup post bytes use approximate packing adjustments. For zkSync, brown is logical capacity and gray is the minimum-packing lower bound; neither is measured logical usage. Window tables use ratios of sums, not averages of these daily ratios.

Restricting the pre window to February 23-March 12 gives three-chain pooled and Laspeyres changes of -8.83% and -4.67%. The four-chain conditional intervals are +7.52% to +13.50% pooled and -15.26% to -2.59% fixed composition. The comparison has nineteen pre days and twenty-nine post days. It is a date restriction, not a verified control for Delta enablement or operator settings.

5.5 Trends and interpretive limits

Appendix B reports, for the three optimistic rollups, a chain-specific log-linear extrapolation fitted only to pre observations. Daily log intensity is regressed on calendar time separately for each chain, using seven, fourteen, or twenty-nine pre days. Post log deviations are evaluated over the first seven days for the seven-day fit and the first fourteen days otherwise. No treatment-effect interpretation or confidence interval is assigned to these extrapolations.

The fitted pre slopes and extrapolated comparisons vary substantially with the fitting span. For Optimism, the fourteen-day pre trend extrapolation implies a post deviation exceeding 100%, while the twenty-nine-day fit implies approximately 12%. This instability is itself informative: a short, steep pre decline is not a credible mechanical forecast far beyond its fitting period. Extrapolated deviations are reported as diagnostics without treatment-effect confidence intervals.

zkSync is omitted from the point-valued trend diagnostics because a capacity series cannot establish logical-pubdata deviations from a pre-period trend. Its daily conditional interval is shown in Figure 2; capacity-based deviations are not treated as observed changes. Compression settings, spam or farming activity, transaction

complexity, and batching can all change measured intensity; the archive does not separately identify their contributions.

6 Aggregate expenditure and baseline sensitivity

Daily expenditure falls from 375.32 to 20.11 ETH, while transactions rise from 2.90 million to 5.49 million. The daily expenditure decline is 94.64%; expenditure per transaction declines 97.17%. The post total is 583.33 ETH in the frozen panel. These are descriptive changes in attributed expenditure and counted activity, not an estimate of expenditure avoided because of Dencun.

The available daily panel begins February 1, 2024. Table 9 reports baseline alternatives within that calendar, with their unequal lengths disclosed.

Table 9 Reproducible baseline sensitivity

Pre dates in 2024	Days	Pre ETH/day	Daily change	Per tx change
02-13 to 03-12	29	375.32	-94.64%	-97.17%
02-01 to 02-12	12	239.43	-91.60%	-96.08%
02-01 to 03-12	41	335.55	-94.01%	-96.94%
02-23 to 03-12	19	427.24	-95.29%	-97.52%

Post window is always March 15–April 12. These alternatives have different durations and can overlap; each uses its own observed pre-period expenditure and transaction totals.

The daily decline is 91.60% against February 1–12 and 94.01% against February 1–March 12. A post-Delta-date baseline is more expensive than the main baseline and gives a 95.29% decline. Thus the main baseline is not the highest-priced feasible subset, and window choice affects the descriptive magnitude in both directions. These comparisons describe sensitivity within the available calendar.

A price-level sensitivity multiplies aggregate expenditure minus blob burn by P_0/P_1 and leaves blob burn unchanged. It gives 28.03 ETH per day and a 92.53% decline against the main baseline. This remains a heuristic because tips and operations outside the selected receipts need not scale with payload-weighted prices. Using the new receipts, repricing the selected execution gas at P_0 gives 200.18 ETH instead of 119.08 ETH. Leaving tips, blob burn, and the aggregate residual at observed values gives 664.44 ETH overall. This is an exact scalar repricing of the selected execution component, with the unclassified residual explicitly held fixed. The main same-payload ratio has no pre-period input and is unaffected by the choice of a pre baseline.

7 Later observations and limits on generalization

Table 10 reports 29-day windows centered 90, 180, and 365 days after March 14, 2024. Aggregate expenditure remains well below the original baseline while activity is higher. These windows are snapshots around each horizon, not cumulative averages or a claim that expenditure was below baseline on every intervening day.

Table 10 Later expenditure and blob prices

Window and dates	Tx/day millions	ETH/day	Blob price
Pre 2024-02-13 to 2024-03-12	2.90	375.32	-
Post 2024-03-15 to 2024-04-12	5.49	20.11	10.112 gwei
90 2024-05-29 to 2024-06-26	6.16	11.26	4.588 gwei
180 2024-08-27 to 2024-09-24	6.93	1.73	1.60 wei
365 2025-02-28 to 2025-03-28	10.57	3.40	2.017 gwei

All windows contain 29 days. Blob price is summed provider fees divided by provisioned blob gas. Later execution-price changes are not separately identified, so no percentage decline is attributed to DA in this table.

The provider-implied blob-gas price differs sharply across these windows: approximately 4.59 gwei at 90 days, 1.60 wei at 180 days, and 2.02 gwei at 365 days. These values are ratios of summed provider blob fees to summed provisioned blob gas, using the four-chain cohort. They illustrate varying price regimes but do not independently validate the provider's fee fields. The archived aggregate series does not supply the payload-weighted execution prices required to distinguish the later execution-price contribution. In particular, its blob-as-calldata fee field must not be mistaken for an execution-price benchmark.

The 90-day window includes the June 20 congestion episode. Resource switching, execution-market conditions, and changing workloads can affect its expenditure. Published accounts of switching are not fully consistent, and this archive contains no complete receipt census for that event. The table therefore makes no claim about continuous blob use or a constant same-payload discount. The 365-day window follows earlier episodes of higher blob demand and should likewise not be characterized as uniformly at the floor.

Later adopters are outside the principal cohort. Their pre-adoption windows can already be exposed to Dencun's effect on execution-market demand, so they would not automatically provide untreated controls.

The historical calldata schedule also cannot be reused automatically under later forks. EIP-7623 specifies a transaction-level calldata cost floor, and EIP-7918 changes the interaction of blob fee adjustment with execution cost. They are boundary conditions for extending the analysis, not inputs to the March-April 2024 charge calculation.

8 Conclusion

For four early blob adopters, the same-inclusion-block calldata benchmark on the recorded physical representation is 11,170.14 ETH against 264.89 ETH of blob burn, independently checked against RPC receipts. The 97.63% charge difference is strongly regime-dependent: nine congestion days account for nearly all sampled blob burn, while the ratio outside that interval is approximately 5.11×10^{-7} .

Decoded payload calldata accounts for 83.78% of aggregate pre-period expenditure. The pre- and post-period receipt tables distinguish payload burn, other execution charges, tips, and unclassified aggregate expenditure. Key-specific nonce counts strengthen posting coverage without assigning residual payments to unobserved operations.

Quantity comparisons depend on the payload representation. The three-chain pooled and fixed-composition ratios decline 10.81% and 6.91% over the principal window. zkSync's post representation measures logical capacity; under minimum packing, adding it gives a four-chain pooled increase of 3.71% to 9.48% and a fixed-composition decline of 6.29% to 18.56%. These intervals describe sampled blob payloads. The four-chain seven-day direction is not identified.

The contribution is descriptive measurement within these cohorts and dates. Provider-derived lengths, the conditional packing rule, differing payload objects, posting lag, and incomplete attribution limit broader interpretation. The findings do not identify causal compression improvements, a no-Dencun equilibrium, or the incidence of lower costs on users, sequencers, or ETH holders.

Generative AI disclosure

Generative AI tools were used for research assistance, code debugging, citation checking, and editorial revision. The author independently reviewed all sources, verified the analysis and reported results, and takes full responsibility for the manuscript.

Appendix A Validation and conditional calldata bounds

The replication package regenerates every table and figure from frozen transaction and daily inputs. It includes the original extraction notebooks, arbitrary-precision audit code, full new RPC receipt responses, complete post-window block base fees, supplemental inputs, and SHA-256 manifests. The accompanying release is an offline replication archive. A permanent public identifier is not yet assigned. The package records the matched growthpie revision, source URLs, new RPC retrieval timestamps, and hashes. The original Blobscan API version and extraction timestamp were not preserved; the archived notebook and input hashes identify the frozen extract without reconstructing those missing historical metadata.

The RPC check validates sampled charges and inclusion prices independently of the original Blobscan fee fields. It does not validate the provider's usage termination rule. A second-archive sidecar sample, matched to versioned hashes and decoded with chain-specific framing, would directly test logical lengths and zero-byte counts. An exhaustive posting-route and selector census would further classify the aggregate residual and possible mixed-resource payloads. The present fee reconciliation does not substitute for either check.

Let R_i be chain i 's nonnegative aggregate expenditure minus all fees paid by its selected blob transactions. If the aggregate completely attributes relevant payer expenditure without offsetting errors, any calldata posting outside the selected receipts must be paid from R_i . Let $p_{\min}$ be a lower bound on its execution base fee and γ a lower bound on intrinsic calldata gas per byte. EIP-2028 gives $\gamma = 4$ without a byte-composition assumption. EIP-3529 permits at most a 20% refund of consumed gas; using $\rho = 0.8$ therefore allows the largest permitted refund (Buterin and Swende 2021). Since paid gas times price is at least $\rho \gamma p_{\min}$ times bytes, an outer bound on additional bytes, u_i , is:

$$u_i \leq R_i / (\rho \gamma p_{\min}). \quad (4)$$

Fees and prices in Equation (4) use the same monetary unit. The independently retrieved minimum across all 206,308 post-window blocks is 7.221509369 gwei. An average price cannot replace that minimum in a hard bound. A compressed-byte scenario with at least 99% nonzero bytes instead uses $\gamma = 15.88$; that is an additional assumption about unobserved postings. Execution overhead and tips in R_i make these bounds loose.

With no byte-composition assumption, the outside-sample byte ceiling is 32.15% of the base upper byte total. Assuming at least 99% nonzero bytes reduces it to 8.10%; additionally assuming every omitted posting paid at least 20 gwei reduces it to 2.92%. Even that last scenario permits a fixed-composition change up to +0.25%, because most residual expenditure is assigned to zkSync, whose pre transaction share is much larger than its post share. The available accounting therefore does not establish the fixed-composition decline's robustness to fallback. Conditional on minimum packing, adding nonnegative omitted post payload raises the pooled lower endpoint. Without that packing assumption, no positive lower-endpoint claim is made.

Table A1 Conditional ceilings on additional calldata

Assumptions	Extra bytes % of base upper	Pooled change upper %	Fixed composition upper %
Any byte composition; 7.22 gwei minimum	32.15	+44.68%	+65.69%
At least 99% nonzero; 7.22 gwei minimum	8.10	+18.34%	+11.84%
At least 99% nonzero; 20 gwei assumed minimum	2.92	+12.68%	+0.25%
Any composition including possible mixed calldata	36.55	+49.49%	+73.64%

Every row assumes aggregate expenditure covers the relevant payer universe without offsetting attribution errors and permits a 20% gas refund. First three rows concern postings outside the selected sample. Last row also treats all calldata that could fit within selected receipt gas as additional payload. These are upper limits, not estimates of actual fallback.

The base four-chain quantity is itself an upper endpoint for zkSync, not a point estimate. Table A1 retains conservative bounds allowing unknown routes beyond the audited keys; the nonce check alone does not cover those routes. “Extra bytes” is expressed relative to the base upper endpoint. For each bound u_i , the pooled upper ratio adds $\sum_i u_i$ to the base upper byte total before dividing by post transactions. The fixed-composition upper ratio adds $\sum_i [(A_{i0} / A_{i1}) u_i] / B_0$ to the base upper Laspeyres ratio, where A_{iw} is chain-window transactions and B_0 is total pre bytes. Chain-specific residuals therefore matter; a pooled percentage cannot simply be added to the fixed-composition change.

The first three rows bound additional postings outside the selected receipts. To cover possible calldata inside sampled blob transactions too, the last row additionally allocates all available sampled receipt gas to hypothetical payload after the standard 21,000 intrinsic gas. For a selected receipt with charged gas g , its input-byte ceiling is $\max[0, g/0.8 - 21,000]/4$. This treats even ordinary call framing as payload and is deliberately loose. All bounds remain conditional on aggregate attribution and comparability of the additional bytes to the measured numerator. They are not estimates of fallback incidence.

Appendix B Local trend diagnostics

Table B1 Pre-only log-trend extrapolation diagnostics

Chain	Pre fit days	Post eval days	Pre log slope % per day	Post deviation %
Arbitrum	7	7	-0.19	-7.13%
Arbitrum	14	14	0.16	-5.95%
Arbitrum	29	14	-0.17	-1.61%
Base	7	7	0.49	-14.43%
Base	14	14	0.15	-9.07%
Base	29	14	-0.40	-1.28%
Optimism	7	7	-1.67	+14.90%
Optimism	14	14	-5.60	+104.22%
Optimism	29	14	-1.38	+11.99%

zkSync is excluded because its post capacity series does not measure logical usage. These are sensitivity diagnostics without confidence intervals or causal interpretation.

For each of Arbitrum, Base, and Optimism, fit $\log(\text{bytes per transaction}) = a + b t$ on pre days only, with t in calendar days relative to March 14. Report $100b$ as the pre log slope and $100[\exp(\text{mean post log residual}) - 1]$ as the deviation from extrapolation. Seven-day fits evaluate seven post days; fourteen- and twenty-nine-day fits evaluate fourteen post days. Equal daily weights differ from the transaction-weighted ratios of sums in the main intensity tables. Appendix C prints intercepts and evaluation means, and the package supplies daily observations and pre sufficient statistics.

Appendix C Inputs for local and composition calculations

For zkSync post rows, the two byte columns give the conditional logical-length lower bound and provisioned logical capacity; all other rows use equal endpoints for the measured proxy. The following inputs reproduce the main, local-window, and post-Delta-date intensity comparisons. Each window uses its own chain transaction shares. The after-Delta-date comparison has nineteen pre days and twenty-nine post days; it is a date restriction rather than a verified control for operator settings.

Table C1 Inputs for 7 days

Chain and period	Bytes lower	Bytes upper	L2 tx	Tx share %
Arbitrum Pre	869,232,773	869,232,773	5,196,941	29.587119
Base Pre	535,300,355	535,300,355	2,975,586	16.940546
Optimism Pre	593,838,641	593,838,641	2,529,617	14.401564
zkSync Era Pre	359,461,048	359,461,048	6,862,733	39.070772
Arbitrum Post	1,469,373,771.781	1,469,373,771.781	9,658,802	29.791349
Base Post	1,550,174,579.945	1,550,174,579.945	9,755,286	30.088942
Optimism Post	1,053,612,994.688	1,053,612,994.688	4,570,416	14.096868
zkSync Era Post	154,534,030	692,654,080	8,436,995	26.022841

Pre: 2024-03-06 to 2024-03-12. Post: 2024-03-15 to 2024-03-21. zkSync post endpoints are conditional minimum-packing length and logical capacity. Other endpoints coincide. CSV values retain full precision.

Table C2 Inputs for 14 days

Chain and period	Bytes lower	Bytes upper	L2 tx	Tx share %
Arbitrum Pre	1,874,973,884	1,874,973,884	11,253,174	29.802221
Base Pre	1,037,251,577	1,037,251,577	5,763,849	15.264627
Optimism Pre	1,663,907,173	1,663,907,173	5,405,128	14.314612
zkSync Era Pre	907,159,799	907,159,799	15,337,364	40.618541
Arbitrum Post	3,321,886,743.469	3,321,886,743.469	20,586,846	30.086271
Base Post	3,433,684,538.867	3,433,684,538.867	20,527,781	29.999952
Optimism Post	2,409,749,234.578	2,409,749,234.578	9,706,382	14.185215
zkSync Era Post	440,486,257	1,267,474,432	17,605,038	25.728562

Pre: 2024-02-28 to 2024-03-12. Post: 2024-03-15 to 2024-03-28. zkSync post endpoints are conditional minimum-packing length and logical capacity. Other endpoints coincide. CSV values retain full precision.

Table C3 Inputs for 29 days

Chain and period	Bytes lower	Bytes upper	L2 tx	Tx share %
Arbitrum Pre	3,939,442,593	3,939,442,593	23,147,497	27.509083
Base Pre	2,264,936,984	2,264,936,984	12,133,703	14.420006
Optimism Pre	3,503,400,571	3,503,400,571	11,199,521	13.309800
zkSync Era Pre	2,717,683,645	2,717,683,645	37,664,202	44.761111
Arbitrum Post	6,969,707,919.086	6,969,707,919.086	44,157,099	27.710086
Base Post	10,610,180,859.820	10,610,180,859.820	61,778,152	38.767898
Optimism Post	5,861,531,718.938	5,861,531,718.938	19,898,462	12.486964
zkSync Era Post	963,885,498	2,320,232,448	33,520,172	21.035052

Pre: 2024-02-13 to 2024-03-12. Post: 2024-03-15 to 2024-04-12. zkSync post endpoints are conditional minimum-packing length and logical capacity. Other endpoints coincide. CSV values retain full precision.

Table C4 Inputs for after Delta date

Chain and period	Bytes lower	Bytes upper	L2 tx	Tx share %
Arbitrum Pre	2,567,087,107	2,567,087,107	15,226,064	27.670824
Base Pre	1,424,465,556	1,424,465,556	7,808,834	14.191249
Optimism Pre	2,235,010,901	2,235,010,901	7,438,841	13.518849
zkSync Era Pre	1,611,288,129	1,611,288,129	24,551,959	44.619078
Arbitrum Post	6,969,707,919.086	6,969,707,919.086	44,157,099	27.710086
Base Post	10,610,180,859.820	10,610,180,859.820	61,778,152	38.767898
Optimism Post	5,861,531,718.938	5,861,531,718.938	19,898,462	12.486964
zkSync Era Post	963,885,498	2,320,232,448	33,520,172	21.035052

Pre: 2024-02-23 to 2024-03-12. Post: 2024-03-15 to 2024-04-12. zkSync post endpoints are conditional minimum-packing length and logical capacity. Other endpoints coincide. CSV values retain full precision.

Table C5 Parameters and post means for trend diagnostics

Chain	Pre days	Intercept a	Slope b	Mean post t	Mean post log intensity
Arbitrum	7	5.11057280	-0.00188569	4.0	5.02904442
Arbitrum	14	5.13059091	0.00164603	7.5	5.08162931
Arbitrum	29	5.11044718	-0.00167209	7.5	5.08162931
Base	7	5.21784211	0.00493014	4.0	5.08169725
Base	14	5.20613413	0.00148919	7.5	5.12223000
Base	29	5.16522990	-0.00401820	7.5	5.12223000
Optimism	7	5.36576505	-0.01665056	4.0	5.43801357
Optimism	14	5.20538161	-0.05599806	7.5	5.49942605
Optimism	29	5.48993170	-0.01383589	7.5	5.49942605

Time t is calendar days from March 14. Post deviation = 100 times $\{\exp(\text{mean post log intensity} - a - b \text{ times mean post } t) - 1\}$. Seven-day fits evaluate March 15-21; fourteen- and twenty-nine-day fits evaluate March 15-28. zkSync is omitted; its daily bounds are supplied separately.

References

Web references were accessed September 13-15, 2026 UTC, except source snapshots explicitly identified as archived. Current documentation is not evidence of historical operator configuration.

- Akhunov, Alexey, Eli Ben Sasson, Tom Brand, Louis Guthmann, and Avihu Levy. 2019. EIP-2028: Transaction Data Gas Cost Reduction. <https://eips.ethereum.org/EIPS/eip-2028>.
- Blobscan. Transaction API fields in the frozen extraction: `usageSize`, `blobAsCalldataGasUsed`, and `blobGasBaseFee`. Extraction notebook included in the replication package. Original API version and extraction timestamp were not recorded. <https://blobscan.com/>. Field semantics are provider-derived and have not been independently raw-body validated.
- Buterin, Vitalik, and Martin Swende. 2021. EIP-3529: Reduction in Refunds. <https://eips.ethereum.org/EIPS/eip-3529>.
- Buterin, Vitalik, et al. 2022. EIP-4844: Shard Blob Transactions. Ethereum Improvement Proposals. <https://eips.ethereum.org/EIPS/eip-4844>.
- Crapis, Davide, Edward W. Felten, and Akaki Mamageishvili. 2023. EIP-4844 Economics and Rollup Strategies. [arXiv:2310.01155](https://arxiv.org/abs/2310.01155). <https://arxiv.org/abs/2310.01155>.
- dRPC. 2026. Ethereum JSON-RPC endpoint. <https://eth.drpc.org>. Receipt and fee-history responses retrieved September 14, 2026 UTC and frozen in the replication package.
- Elowsson, Anders, Ben Adams, and Francesco D'Amato. 2025. EIP-7918: Blob Base Fee Bounded by Execution Cost. <https://eips.ethereum.org/EIPS/eip-7918>.
- Ethereum Execution APIs. 2026. `eth_getTransactionReceipt` and `eth_feeHistory`. https://ethereum.github.io/execution-apis/api/methods/eth_getTransactionReceipt/; https://ethereum.github.io/execution-apis/api/methods/eth_feeHistory/.
- Ethereum Foundation, Protocol Support Team. 2024. Dencun Mainnet Announcement. February 27. <https://blog.ethereum.org/2024/02/27/dencun-mainnet-announcement>.
- growthepie. 2026. Backend Metric Configuration and Transaction-Count Queries. Commit `2bea111699a0cb38e17f1a0c5f4897ef976fbb9d`. <https://github.com/growthepie/gtp/tree/2bea111699a0cb38e17f1a0c5f4897ef976fbb9d>. Archived files match this revision; historical execution provenance remains unverified.
- Heimbach, Lioba, and Jason Milionis. 2025. The Early Days of the Ethereum Blob Fee Market and Lessons Learnt. [arXiv:2502.12966v1](https://arxiv.org/abs/2502.12966v1). <https://arxiv.org/abs/2502.12966v1>.
- Huang, Yue, Shuzheng Wang, Yuming Huang, and Jing Tang. 2024. Two Sides of the Same Coin: Large-scale Measurements of Builder and Rollup after EIP-4844. [arXiv:2411.03892v1](https://arxiv.org/abs/2411.03892v1). <https://arxiv.org/abs/2411.03892v1>. Later versions have a different title and author list.
- Mamageishvili, Akaki, and Edward W. Felten. 2023. Efficient Rollup Batch Posting Strategy on Base Layer. Workshop on Trusted Smart Contracts. https://fc23.ifca.ai/wtsc/WTSC23_1.pdf.
- Marshall, Blair. 2024. Blobsplaining Part 2: Lessons From The First EIP-4844 Congestion Event. Blocknative, April 9. <https://www.blocknative.com/blog/blobsplaining-part-2-lessons-from-the-first-eip-4844-congestion-event>.
- Matter Labs. 2024. ZKsync Era KZG Encoding. Repository revision `0764227aa22165cba5e3d41756383cfe940e97b4`, March 14. `core/lib/l1_contract_interface/src/i_executor/commit/kzg/mod.rs`. <https://github.com/matter-labs/zksync-era/tree/0764227aa22165cba5e3d41756383cfe940e97b4>. Frozen source in the replication archive; source availability does not independently prove historical binary deployment.
- Offchain Labs. 2024. Nitro Blob Encoding. Repository revision `b51bf5641d36123ff70efa7f3261b8cb3e2ab3c0`. `util/blobs/blobs.go`. <https://github.com/OffchainLabs/nitro/tree/b51bf5641d36123ff70efa7f3261b8cb3e2ab3c0>.
- OP Labs. 2024. OP Stack Blob Encoding. Repository revision `804a0997451c6d0dcee701a4f9c2679242fca04c`. `op-service/eth/blob.go`. <https://github.com/ethereum-optimism/optimism/tree/804a0997451c6d0dcee701a4f9c2679242fca04c>.
- Palakkal, Roshan, Jan Gorzny, and Martin Derka. 2024. SoK: Compression in Rollups. IEEE International Conference on Blockchain and Cryptocurrency, 712-728. <https://doi.org/10.1109/ICBC59979.2024.10634469>.
- Sunnyside Labs. 2024. How the Delta Upgrade Reduced OP Chain Fixed Costs by over 90%. Optimism, February 29. <https://optimism.io/blog/how-the-delta-upgrade-reduced-op-chain-fixed-costs-by-over-90>.
- Test in Prod. 2024. Upgrade Proposal 3: Delta Network Upgrade. Optimism Collective governance. <https://gov.optimism.io/t/final-upgrade-proposal-3-delta-network-upgrade/7310>.
- Ved, Tanay, and Matias Andrade. 2024. Breaking Down Ethereum Blobs & EIP-4844. Coin Metrics State of the Network 262, June 4. <https://coinmetrics.substack.com/p/state-of-the-network-issue-262>.
- Wahrstätter, Toni, and Vitalik Buterin. 2024. EIP-7623: Increase Calldata Cost. <https://eips.ethereum.org/EIPS/eip-7623>.