



A settlement layer for the agent-paid web.

Make the smallest unit of value sellable — a read, a second, a citation, a listen, a call — metered and settled on Arc in USDC. For humans and machines.

ARC · CIRCLE GATEWAY · X402 · ERC-8004

Testnet reference implementation — sluiceflow.vercel.app

Contents

Abstract	3
1. The problem: read everywhere, paid nowhere	3
2. The gap x402 leaves: deferred, aggregated settlement	3
3. The Meter: one mechanism, every unit	4
4. Settlement: gas-free, batched, anchored	4
5. Proof-of-Flow: honest streaming	4
6. The Citation Toll: payment is the citation	4
7. Reputation as capital at risk	5
8. Quadratic funding for the long tail	5
9. Architecture & developer surface	5
10. Deployment status (July 2026)	5
11. Conclusion	6
References	6

Abstract

The open web is being read, at scale, by machines that rarely pay or refer traffic back. New standards have emerged to declare terms — Cloudflare's Pay Per Crawl and the Really Simple Licensing (RSL) standard — but declaring a price is not the same as collecting it. Sluice is the missing settlement layer: it wraps any resource behind an x402 paywall, meters usage in any unit, and settles batched nanopayments gas-free through Circle Gateway on Arc. Payment and attribution become the same event. Reputation becomes capital at risk, recorded on-chain. The result is an open toll booth for the long tail — creator-owned, agent-payable, and verifiable.

1. The problem: read everywhere, paid nowhere

By 2025, AI crawlers had become a dominant source of automated traffic, while sending comparatively little referral traffic back to the sites they learn from. Cloudflare's own 2025 reporting highlighted how lopsided the crawl-to-referral relationship had become for AI bots relative to traditional search — orders of magnitude more pages scraped than visitors returned. For a creator, the historical bargain of the open web (crawl me, and send me readers) is quietly dissolving.

The ecosystem responded with mechanisms to state terms. Cloudflare's Pay Per Crawl (introduced 2025) lets publishers charge AI crawlers per request using the HTTP 402 'Payment Required' status — but it settles only within Cloudflare, for Cloudflare's customers. The Really Simple Licensing standard (RSL 1.0, 2025), backed by major publishers, extends robots.txt and sitemaps with machine-readable licensing and royalty terms — free, subscription, per-crawl, or per-inference — but it is a terms layer, not a payments rail.

These are real progress, but they largely standardize the declaration of terms. The harder half is settlement: collecting micro-amounts, across many counterparties, at sub-cent granularity, without gas eating the payment — and doing it for machines that transact autonomously. That is the gap Sluice fills: an open toll booth that any creator can own, any agent can pay, and anyone can audit.

2. The gap x402 leaves: deferred, aggregated settlement

x402 is an elegant handshake: request a resource, receive HTTP 402 with a price, sign a payment authorization, retry, get the content. It answers 'how does a machine pay for one thing, once.' It does not answer how to charge for the ten-thousandth of a stream-second, the third citation in an answer, or a crawl priced below what any single settlement can economically carry.

Sluice's core primitive — the Meter — decouples authorization from settlement. A payer authorizes once; consumption accrues per unit in integer base units, including amounts below the settleable floor; and accruals are aggregated and settled in batches through Circle Gateway, gas-free, on a timer. The unit adapter is pluggable (per request, per read, per crawl, per citation, per second, per listen), and the settlement backend is swappable (Gateway-batched primary, direct x402 fallback) without touching accrual logic. Metering is the product; settlement is a backend.

```
rate × units → accrued (6dp base units, bigint)
batch → Circle Gateway → settled receipt (attested) → withdraw on-chain
```

3. The Meter: one mechanism, every unit

Sluice meters value in whatever unit a resource is naturally consumed: per request, per read, per crawl, per citation, per second, per byte, per token, per listen, per view. A payer signs an x402 authorization against the Gateway Wallet; usage accrues per unit and is batched into settlement. A resource declares its unit and rate once; everything downstream — accrual, batching, receipts, badges, payouts — is shared machinery.

Decimals discipline

Payment USDC is six-decimal and handled exclusively as integer base units — never floating point. Native Arc gas is USDC displayed at eighteen decimals and tracked separately. The two are never coerced into one another, eliminating an entire class of rounding and precision bugs. Money crosses JSON boundaries as strings and is parsed back to bigint at the edge.

4. Settlement: gas-free, batched, anchored

The payment handshake follows the Gateway Nanopayments profile of x402. A 402 response carries the price and an EIP-712 domain whose verifying contract is the Circle Gateway Wallet — the buyer signs an EIP-3009 TransferWithAuthorization against the Gateway Wallet contract, not the USDC token. Circle Gateway verifies the signature in under a second, the resource is served immediately, and settlement is deferred into batches.

Circle Gateway settles those batches via an attested ledger, so individual nanopayments do not each burn gas. Each settlement carries a Circle transfer identifier rather than a per-payment transaction hash. Funds touch the chain at well-defined anchors: the depositor's one-time deposit into the Gateway Wallet, and the recipient's withdrawal — a Gateway Minter mint, instant on Arc or cross-chain. Royalty splits, reputation bonds, and funding-round sweeps are additional on-chain anchors. All are independently verifiable on Arcscan.

Authorizations are signed with long validity windows so batched settlement remains valid; the platform settles on a timer and a reconciler resolves each Circle transfer to a confirmed receipt. The UI never shows 'settled' before the batch settles: states are authorized, then batching, then settled — honestly.

5. Proof-of-Flow: honest streaming

For continuous resources, the payer approves a rate and a reserve cap. The meter accrues lazily — frozen accrued time plus a live flowing delta, capped at the reserve. A lightweight heartbeat proves delivery. If the heartbeat goes stale, the meter auto-pauses and freezes accrual at the last good heartbeat, not at detection time, so dead air is never billed. When delivery resumes, so does the meter. On stop, only the flowed whole seconds settle; unused reserve is simply never charged.

6. The Citation Toll: payment is the citation

When a research agent grounds an answer on a source, it pays that source to retrieve it. The payment is the citation — the same event serves attribution and compensation, and both are auditable: claim, source, payment, author — one chain. Single-author resources settle gas-free via Gateway. Multi-author resources deploy a per-resource RoyaltySplitter contract that fans the

payment out by share on-chain, with the final payee absorbing rounding dust so the split is exact. This bridges the terms layer to the money layer: a resource's RSL document and llms.txt declare the license and the price; Sluice is the mechanism that actually collects it when an agent honors those terms. Creators get an embeddable badge whose earned counter is computed from real settled receipts.

7. Reputation as capital at risk

Star ratings are cheap to fabricate; staked capital is not. In Sluice, a provider self-bonds USDC behind a job. On delivery the bond is released; on underdelivery an arbiter slashes it to the harmed buyer. A minimal ERC-8004 Identity and Reputation registry pairs each provider with an on-chain identity and records feedback on resolution. The bond contract's running totals — bonded, active, slashed, released — are the reputation: a fact you can read as money rather than a score you must trust.

8. Quadratic funding for the long tail

Toll revenue rewards what agents already consume; it does not bootstrap what should exist next. Sluice adds an on-chain funding pool with quadratic matching: many small backers move the match more than one large one ($\text{match} = (\text{sum of square roots of contributions})^2$, minus the contributions themselves), with a conservative sybil weighting derived from on-chain identity age and activity. Tips are real transfers; the round budget is committed up front; and the round settles in a single on-chain sweep from the FundingPool contract.

9. Architecture & developer surface

- Registry + paywall: any resource becomes an x402-protected, priced endpoint.
- The Meter: unit adapters + accrual ledger, decoupled from the settlement backend (Gateway-batched primary, direct x402 fallback).
 - @sluice/pay SDK: one-call payment, deposit-aware, with budget and reasoning hooks.
 - MCP server: discover / price / pay / receipts / register as native agent tools.
 - RSL + llms.txt: declares terms in the emerging standards; Sluice is the settlement that honors them.
 - Buyer agent: an LLM scores relevance and recommends; deterministic code enforces budget, price ceilings, and allowed units. Model output never authorizes a payment.
 - Self-hostable toll sidecar: creator-owned, settle to your own wallet, no platform lock-in.

```
const sluice = new SluicePay({ privateKey });
const { formattedAmount } = await sluice.pay(resourceId, { maxAmount: 0.01 });
```

10. Deployment status (July 2026)

Sluice runs live on Arc testnet (chain id 5042002) at sluiceflow.vercel.app. Every figure shown in the product traces to a real on-chain event or a real database record; there is no simulated data. The following contracts are deployed and verified on Arcscan (testnet.arcscan.app):

```
IdentityRegistry      0x8e856716d653db35eb4dac7616648172cebeba34
ReputationRegistry   0x6593cd1eb1dec37797aee650d48ad2f4d910cbd4
BondEscrow           0x1bf29623c8a74c13bc4e27bbe72037a24976c0c1
FundingPool          0xf7ef1d456e74736bbf346c29f74e28c60ce3ade8
RoyaltySplitter      deployed per multi-author resource
Circle Gateway Wallet (Arc testnet) 0x0077777d7EBA4688BDeF3E311b846F25870A19B9
```

Verified end-to-end on testnet: deposits, sub-cent citation tolls, streaming sessions with proof-of-flow, on-chain royalty splits, bond post/slash/release with ERC-8004 feedback, quadratic-round sweeps, and real same-chain and cross-chain withdrawals via the Gateway Minter. Mainnet posture: Arc mainnet is announced for late 2026; Sluice's chain access is confined to one network-agnostic interface, so redeployment is a configuration change plus contract redeploys — not a rewrite.

11. Conclusion

Standards now let the web declare what access costs. Sluice makes that price collectible — at sub-cent granularity, gas-free, for humans and autonomous agents alike, with attribution and reputation built in. It turns the long tail of content into an addressable, payable surface, and gives the agent economy a settlement layer it can build on.

References

- [1] Cloudflare. Introducing pay per crawl: enabling content owners to charge AI crawlers (2025).
- [2] Cloudflare Radar. AI bot crawl-to-referral ratios, 2025 reporting.
- [3] RSL Collective. Really Simple Licensing (RSL) 1.0 standard — rslstandard.org (2025).
- [4] Circle. Gateway documentation — developers.circle.com/gateway (incl. Gateway Nanopayments).
- [5] x402 — an open payments standard over HTTP 402 — x402.org.
- [6] EIP-3009: Transfer With Authorization (USDC's meta-transaction standard).
- [7] ERC-8004: Trustless Agents — Ethereum ERC draft (2025).
- [8] Circle. Arc — an open Layer-1 for stablecoin finance — docs.arc.network.