TapeMine: Entitlements and Revenue Allocation for Community Mining

TapeMine

Technical Whitepaper | Reference implementation: TapeoutMinerPool/2.3.0

Abstract

TapeMine converts community contributions of materials in the TapeOut circuit-mining ecosystem into transferable revenue entitlements. Participants contribute NAND and LATCH; the operator reconciles contributions and organizes miner construction; entitlement receipts record participants' rights within each processor and material class. Each processor has an Optimal Pool and an Ordinary Pool. Both pools recognize the same receipts while separately accounting for BEM actually received and per-receipt payment history. The design combines fixed receipt supplies, fixed top-level allocation weights, atomic miner admission, checkpoints for changes in material composition, and payment tracking by receipt ID. These mechanisms constrain post-initialization rule changes and prevent duplicate payment of recorded entitlements. In each pool, ordinary revenue is allocated between classes in proportion to cumulative material consumption. A newly admitted miner's carry-in revenue is attributed according to its own material composition; changes in pool composition do not recalculate historical class entitlements. This paper describes the lifecycle, allocation model, worked examples, and verification conditions, together with limitations arising from early operational custody, permanent miner custody, upstream dependencies, and operating capacity. It addresses the 2.3.0 candidate implementation. Deployment and review status are tied to an identified source snapshot.

Keywords: TapeMine; TapeOut; community mining; ERC-1155; revenue allocation; smart contracts

1 Introduction

In the TapeOut ecosystem, NAND and LATCH must be consumed to construct circuit assets eligible for mining before they can earn BEM through the external Mining protocol. Material holders may lack the time or tools required for circuit design, construction, and ongoing operation. Pooling materials enables the community to coordinate this work, but also creates a need to allocate the revenue of jointly constructed miners.

TapeMine treats material contributions, entitlement receipts, and miner custody as separate components. The operator reconciles incoming transfers and coordinates construction; receipts record participants' revenue entitlements; and custody pools distribute the BEM they actually receive according to public rules. Each processor has an Optimal Pool and an Ordinary Pool. The two pools share one set of receipts but maintain separate revenue accounts and claim histories.

The architecture imposes constraints at issuance, initialization, admission, and payment. Receipt supply cannot change after issuance finalization. The relevant initialization owners must renounce ownership before miners enter a production pool. When the material composition changes, checkpoints preserve the entitlements previously accrued to each class. Amounts already claimed are recorded by pool and receipt ID. These constraints remain conditional on correct reconciliation, deployment configuration, and external protocol behavior.

This paper examines the TapeoutMinerPool/2.3.0 candidate implementation and its associated operating policies. The cited official whitepaper, version 1.0, dates its source snapshot to November 1, 2026; Section 10 describes the deployment and review status of the reference version. Source-code capabilities, operating policies, and evidence of actual deployment are distinguished throughout. TapeMine runs on BNB Smart Chain. It does not issue BEM, introduce a new governance token, or alter the issuance rules of the upstream Mining protocol. [1,4,5]

2 System Model and Asset Definitions

2.1 Processors and Assets

TapeOut denotes both the circuit-mining ecosystem and one of its processors; the other processor is Behemoth. Each processor has its own material collection, miner collection, receipts, distributor, and custody pools. Materials and entitlements are not shared across processors. BEM is an external ERC-20 reward token with 8 decimal places: 1 BEM equals 10^8 smallest units.

Table 1. Assets and Entitlement Objects in the System

Object	Definition	Technical Representation
NAND / LATCH	Materials used to construct circuits; issued by the upstream protocol, not by TapeMine.	ERC-1155; material IDs 0 / 1
Miner	A circuit asset constructed by consuming materials, with mining eligibility determined by the Mining protocol.	Upstream ERC-721 NFT
Entitlement receipt	A unit of entitlement to revenue from both pools of its processor under fixed rules.	New ERC-1155 collection; supply of 1 per ID
BEM	The reward token paid by external Mining and the unit used for pool accounting and payments.	ERC-20; 8 decimal places

Each processor deploys two entitlement receipt collections, one for NAND and one for LATCH. IDs within each collection run consecutively from 1 to the final supply, with a supply of 1 for each ID. NAND receipt #1 and LATCH receipt #1 belong to different collections. A receipt entitles its holder to a share of its processor's pool revenue under fixed rules; it confers neither a right to redeem the original materials nor fractional ownership of a specific miner.

2.2 Two-Pool Structure

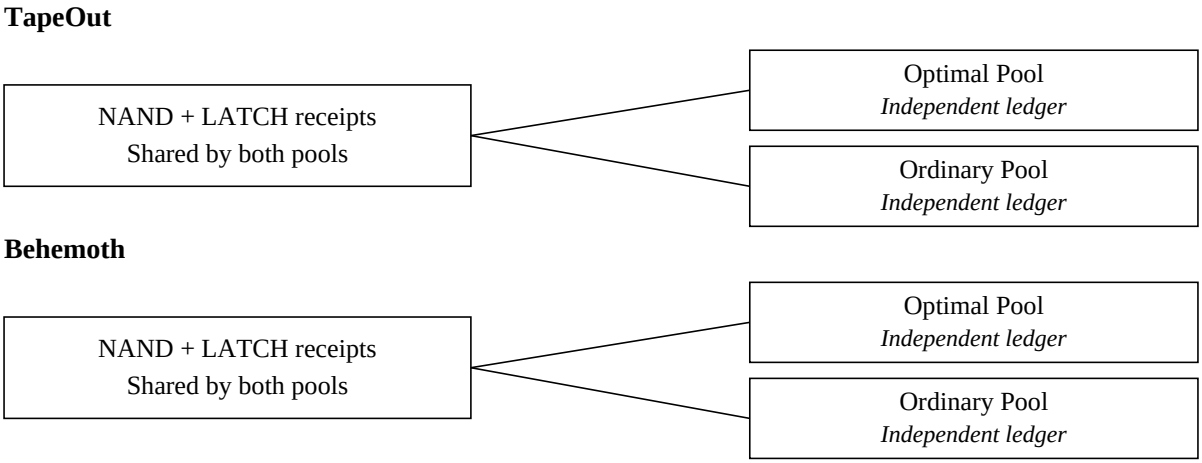


Figure 1. Independent Receipt and Two-Pool Structures for the Two Processors

Each processor uses two receipt contracts, one distributor, and two custody pools, for a total of five stateful instances. Deploying both processors requires ten instances. The Optimal Pool and the Ordinary Pool recognize the same receipts, which are issued, allocated, and delivered only once. Each pool maintains its own miner registry, BEM balance, class ledger, and claim history. [1,5]

3 Contributions, Receipt Issuance, and Allocation

3.1 Historical Contribution Round and Eligibility Records

Table 2. Historical Contribution Round and Eligibility Rules

Historical Round	Published Rules
Network	BNB Smart Chain, chain ID 56.
Displayed opening time	October 20, 2026, 08:00 UTC.
Eligibility cutoff	October 25, 2026, 08:00 UTC; the round has ended.
Early and late transfers	The source documents state that eligible transfers received before opening are also counted. Transfers received at or after the cutoff remain visible in the history but do not qualify for this round.
Basis of registration	Sender address, processor, material class, and quantity; the final public reconciliation and on-chain allocation must be checked against each other.

The public contribution wallet is 0x8232b40d5b375db47504e56082b11d8d016b22bb. The website's time window does not prevent the blockchain from accepting direct transfers. Sending materials after the round has closed does not establish eligibility for this round.

For example, under the participation policy, 10 eligible units of Behemoth NAND correspond to 10 Behemoth NAND entitlement receipts. They do not represent 10 miners or 10% of pool revenue. Each receipt represents one unit of entitlement within the same processor and material class.

Materials are contributed through an ordinary transfer of the upstream ERC-1155 tokens. This transfer does not simultaneously call a TapeMine contract, reserve receipt IDs, or immediately mint

receipts. The operator implements the 1:1 policy through issuance and allocation; the contracts themselves cannot verify that the historical contribution records are complete.

The quantity of contributed materials determines the expected number of receipts, but does not directly determine the revenue split between the NAND and LATCH classes. Eligible contributions are registered to the original sending address. Until the separate issuance and allocation processes are complete, the operator handles the materials and records, and contributors depend on custody and reconciliation during this stage.

3.2 Receipt Issuance and Fixed Supply

Receipt IDs in each class run consecutively from 1 to the final supply S, with a supply of exactly 1 per ID. The issuer may call issue in batches. Before finalization, retract(count) can burn only the highest receipt IDs, all of which must still be held by the issuer; it cannot confiscate receipts held by others.

finalizeIssuance() is irreversible. After it is called, all minting and burning are rejected, while transfers remain available. Before owner renunciation, the issuer may change the metadata URI but cannot override the finalized supply. Adding miners later does not issue new receipts for this round.

3.3 Allocation Records and Finalization

The distributor accepts only the two bound receipt classes and rejects receipts with a claim history in either pool. The tooling converts a file of (wallet, nandCount, latchCount) entries into ID ranges, assigning the lowest available IDs in the order of wallets in the file. allocate requires each ID to be held by the distributor and not yet allocated. The beneficiary cannot be the zero address, the distributor, or either pool.

Before finalization, revokeWallet can revoke all allocations to a wallet. finalizeAllocation() requires issuance to be finalized for both classes and all receipts currently held by the distributor to be allocated. After finalization, beneficiaries and ID ranges cannot be changed, and the distributor no longer accepts receipts.

Complete allocation of the distributor's inventory does not establish that every issued receipt entered the distributor or that every eligible contributor is covered. Eligible contributions, supply, inventory, and beneficiary records still require item-by-item reconciliation.

3.4 Receipt Delivery

A beneficiary may call claim(maxIds), and anyone may call deliver(wallet, maxIds). Both routes send receipts only to the recorded beneficiary. Each call is subject to a quantity limit, each ID can be delivered only once, and there is no claim deadline. Large allocations may require multiple transactions.

Delivery uses an ERC-1155 batch transfer. An incompatible contract wallet may be permanently unable to receive its receipts, and delivery cannot be redirected to another address. The distributor does not claim BEM, approve operators, or make state-changing calls to either pool.

Delivery and mining operations need not occur at the same time. Once allocation is finalized, receipts may be delivered according to the records. Receipts still held by the distributor also accrue entitlements when the pools generate revenue. Delivery has no deadline, and delayed delivery does not erase accrued, unpaid amounts. Compatibility with ERC-1155 batch transfers should be verified before finalization.

3.5 Reconciliation and Exception Handling

Public visibility of the contribution wallet does not mean that materials in the preliminary stage are already subject to the production pools' permanent custody restrictions. The source documents do not require a multisignature arrangement for initialization, and its use cannot be assumed. The operator still needs to publish the final reconciliation, objection and correction deadlines, procedures for late or erroneous transfers, treatment of unused materials, losses from failed construction, and responsibility for gas costs. The 1:1 receipt allocation policy must be verified by reconciling eligible contributions, total issuance, inventory, and beneficiary records item by item. [1-3,8]

4 Miner Construction and Custody

4.1 Miner Construction and Pool Assignment

The project plans to prioritize the construction of miners that compete for optimal slots, then use the remaining materials to construct ordinary miners. The operator is responsible for circuit search, material planning, construction, and pool assignment. The available public materials do not provide a complete, reproducible optimization algorithm or guarantee the number of optimal miners or their future output.

Table 3. Pool Assignment and External Miner Status

Status or Name	Meaning	Effect on Revenue and Custody
Optimal Pool	Holds miners constructed with the aim of competing for optimal slots.	The name does not establish continuing optimal status. The contract does not check whether a miner currently occupies an optimal slot.
Ordinary Pool	Holds ordinary miners constructed from remaining materials under the operating plan.	Uses the same contract rules as the Optimal Pool; its revenue is not guaranteed to be lower.
Loss of an optimal slot	According to the cited investigation, the miner remains ACTIVE, but its optimal-status bonus is removed and its weight is recalculated.	May continue earning revenue; remains in its original custody pool and does not automatically move to another pool.
REVOKED	The circuit has been permanently revoked following a valid challenge.	Future output stops, but the miner remains in the pool. Its consumed materials continue to count toward the class proportions.
COOLING	The external protocol's status after a miner is stopped.	The production pool does not call stop, so its normal operation should not cause the miner to enter this status.

Pool names indicate operational assignment; the contracts do not use them to verify that miners continuously hold optimal slots. Allowlisted depositors decide which pool receives an eligible miner. A miner admitted to the Optimal Pool remains there after losing its optimal slot. The Ordinary Pool and the Optimal Pool use the same code, and their relative revenues are not guaranteed.

According to the project's prior investigation, a valid challenge first pays accrued revenue to the miner's current holder and then permanently revokes the miner. When the holder is a pool, that payment enters the pool and is recorded during subsequent revenue recognition. This finding depends on the

version investigated; the current implementation must be checked at a specified block before production launch.

4.2 Atomic Admission Procedure

Only allowlisted depositors may call depositMiners(tokenIds). The caller must own the miners and authorize the pool to transfer them. Each batch contains 1 to 50 IDs, with no duplicates or previously registered IDs. Miners must be ACTIVE, their collections and IDs must match, and their consumption records must agree with the cached values and have a nonzero total. A miner cannot consume materials from a class for which no receipts were issued.

Table 4. Transaction Sequence for Supported Miner Admission

Step	Actions Within One Admission Transaction
1	Validate the batch and miner materials; on first admission, also check finalization and ownership status.
2	Recognize newly received BEM already in the pool using the previous material composition.
3	Settle all previously registered miners and record the proceeds under the previous composition. Skip REVOKED miners; any other claim failure reverts the admission.
4	Save a checkpoint for the previous period, preserving accrued class entitlements, including fractional amounts, then receive the new miners.
5	Claim each new miner's carry-in revenue separately and attribute it according to that miner's own material consumption proportions.
6	Add the new miners' consumption to the pool-wide composition. Subsequent ordinary revenue uses the new proportions.

If any step fails, the entire transaction reverts and the miners remain with the depositor. Before first admission, the pool has no material composition and recognizes no revenue. First admission processes the new miners' carry-in revenue before recognizing any other previously deposited BEM as ordinary revenue for the new period.

4.3 Direct Transfers and Permanent Custody

registerMiners remains available as an entry point but is not a supported production path. Between an ordinary transferFrom and registration, another party may trigger a claim for the miner, with the revenue recorded under the composition then in effect. If the miner is revoked during that interval, it may become permanently ineligible for registration while remaining in the pool. A direct transfer does not automatically register a miner. The pool has no ERC-721 receiver hook, so safeTransferFrom reverts.

Production pools have no withdrawal, emergency return, migration, asset sweep, pause, upgrade, or arbitrary-call entry points. Admitted miners cannot move between pools, and there is no general recovery mechanism for incorrect pool assignments or mistakenly transferred NFTs. The repository's recoverable canary contract is for testing only and is not a production feature. [1,5,7,10]

5 Revenue Recognition and Top-Level Revenue Allocation

5.1 Recognition of Received Revenue

Anyone may call harvest(ids) or harvestRange(start,count). These functions claim from Mining for each miner and isolate individual miner failures. sync() is also permissionless. Unlike the settlement of all existing miners during admission, ordinary harvesting can continue processing other miners when a claim for one miner fails.

$$\text{received} = \text{balance} - (T - \text{totalPaid}) \tag{1}$$

T is cumulative recognized revenue and totalPaid is the cumulative amount paid; all values are expressed in BEM smallest units.

BEM arriving at a pool is generally recognized as ordinary revenue for the current period, including miner claims, Mining.claim calls triggered directly by third parties, and ordinary BEM transfers. Carry-in revenue measured for each new miner during admission is instead attributed separately using that miner's own material composition. Estimates returned by Mining.pending() are not recognized as received revenue.

An entitlement claim first synchronizes newly received BEM already held by the pool, then reads the ledger and holder index and makes the payment. The claim function itself does not call Mining. Consequently, provided that conditions such as normal BEM transfers and a sufficient balance hold, a Mining outage or failed admission does not necessarily prevent claims against revenue already received.

5.2 Top-Level Revenue Allocation

Table 5. Top-level revenue allocation of BEM revenue in each pool

Destination	On-chain Weight	Share of Total Revenue	Example: 1,050 BEM
Participant entitlement receipts, in aggregate	80 / 84	Approximately 95.2381%	1,000 BEM
Role A: TapeMine-designated role	1 / 84	Approximately 1.1905%	12.5 BEM
Role B: team-designated recipient	1 / 84	Approximately 1.1905%	12.5 BEM
Fixed sink address C	2 / 84	Approximately 2.3810%	25 BEM

The publicly presented ratio of 100:1.25:1.25:2.5 is equivalent to the on-chain ratio of 80:1:1:2. The aggregate non-participant share is 5/105 of total revenue, approximately 4.7619%. A stated “5%” relative to the participant share is therefore not a deduction of exactly 5% from total revenue.

$$A(T) = \left\lfloor \frac{T}{84} \right\rfloor, \quad B(T) = \left\lfloor \frac{T}{84} \right\rfloor$$
$$C(T) = \left\lfloor \frac{2T}{84} \right\rfloor, \quad P(T) = \left\lfloor \frac{80T}{84} \right\rfloor \tag{2}$$

Each term is rounded down independently from cumulative T. P is the cumulative participant share; a payment equals the cumulative entitlement less the amount already paid. For example, when T is 42

smallest units, A and B are both 0, C is 1, and P is 40. Thus C=2A cannot replace the independent calculation. Each pool applies this rule separately. [1,6]

5.3 Recipient Roles and the Sink Share

The weights are code constants. The contract has no entry point to adjust the fee rate or introduce additional contract fees. Gas for on-chain transactions is incurred separately and is not part of the revenue allocation above.

C is fixed at 0x000000000000000000000000000000000000dEaD. Whenever anyone triggers a fee payment to A or B, the unpaid amount for C is also transferred to that address. Until payment occurs, the C share remains in the pool. This is an ordinary BEM transfer: it does not call burn and does not reduce totalSupply. It is treated as removed from circulation only in the conventional sense that no party is known to control the address.

A is the role designated by TapeMine, and B is the recipient designated by the team; the contract does not establish their actual identities. Only the current holder of each role may use setRecipient to change that role's recipient address, and its accumulated unpaid amount moves with the role. Anyone may call releaseA or releaseB to pay the corresponding role and C. This does not claim on behalf of the other role or change its paid record.

Payment to the selected A/B role and payment to C are executed in the same transaction. Even if the selected A/B role currently has nothing due, the call can still pay C's unpaid share. Any failed transfer that is actually attempted, or any unexpected balance change, reverts the transaction. The C address is fixed and cannot be changed. Entitlement claims use a separate path. If an A/B recipient loses control of its address, its share may become inaccessible, but the shares of other accounts do not change. [1,6,7]

6 Class Accounting and Receipt Entitlements

6.1 Class Allocation and Receipt Denominators

Within each period of unchanged material composition, the participant share of ordinary revenue is allocated between classes in proportion to the cumulative NAND and LATCH consumption recorded for all miners admitted to that pool. A new miner's carry-in revenue is attributed separately according to its own material composition. Each class's cumulative entitlement is then divided equally among its fixed number of entitlement receipts. Neither the contribution wallet balance nor the market prices of materials are used to calculate the class proportions.

Table 6. Principal accounting variables

Symbol	Definition and Scope
T	Cumulative recognized revenue of a pool, in BEM smallest units, including carry-in revenue; maintained independently for each pool.
N, L	Total NAND and LATCH consumption of all miners admitted to this pool, including miners subsequently revoked.
S_N, S_L	Final supplies of the processor's two classes of entitlement receipts; both pools use the same denominator for each class.
I_N, I_L	Cumulative revenue of the two classes in a pool, expressed in BEM smallest units.

Table 6. Principal accounting variables

Symbol	Definition and Scope
q_N, q_L	Cumulative entitlement of one entitlement receipt in the corresponding class; not its current unclaimed amount.
claimed[p][c][i]	Amount already paid by pool p for class c, ID i.

$$R_N = R\frac{N}{N+L}, \qquad R_L = R\frac{L}{N+L} \tag{3}$$

R is the ideal participant share of ordinary revenue during a period of unchanged material composition, excluding separately attributed carry-in revenue. R_N and R_L are the amounts allocated to the two classes, with N+L>0. On-chain results are governed by checkpoints and integer arithmetic.

Contributed quantities determine the receipt supplies S_N and S_L, while the material consumption of admitted miners determines class revenue. Even if NAND receives a larger class share, revenue per NAND entitlement receipt may be lower than revenue per LATCH entitlement receipt if the NAND receipt supply is larger. Within the same processor and the same pool, entitlement receipts of the same class have equal cumulative entitlements; their unclaimed amounts depend on each ID's claim history.

6.2 Worked Example: Two Pools

Assume that each class has 100 entitlement receipts, both pools receive only ordinary revenue, material composition remains unchanged, and the amounts divide exactly. Table 7 shows the calculations for each pool. The figures are hypothetical illustrations of the rules, not observed or forecast returns.

Table 7. Revenue example for the same entitlement receipts across two pools

Item	Optimal Pool	Ordinary Pool
Cumulative gross revenue actually received	1,050 BEM	2,100 BEM
Participant share	1,000 BEM	2,000 BEM
Pool material consumption, N : L	3 : 1	1 : 3
NAND class revenue	750 BEM	500 BEM
LATCH class revenue	250 BEM	1,500 BEM
Cumulative entitlement per NAND receipt	7.5 BEM	5 BEM
Cumulative entitlement per LATCH receipt	2.5 BEM	15 BEM

Each NAND entitlement receipt has a cumulative entitlement of 7.5+5=12.5 BEM, and each LATCH entitlement receipt has a cumulative entitlement of 2.5+15=17.5 BEM. Claims deduct the amounts already paid in each pool. The pools cannot first combine their material proportions and then allocate revenue: doing so would allow one pool's material composition to alter the attribution of the other pool's revenue.

6.3 Checkpoints and Carry-in Revenue

A new miner may change the pool-wide NAND-to-LATCH ratio. Recalculating all historical revenue using the new ratio could redistribute entitlements already accumulated in the previous period. TapeMine establishes a checkpoint at the admission boundary: it first settles existing miners and preserves the previous period's class entitlements, then brings the new ratio into effect.

For the following example, assume there is no other prior revenue. The existing pool has cumulative consumption of 9 NAND and 1 LATCH, and the new miner consumes 1 NAND and 9 LATCH. The existing pool has 105 BEM awaiting settlement, and the new miner actually brings in 105 BEM at admission. Both amounts are gross revenue before the top-level revenue allocation.

Table 8. Settlement of the previous period and attribution of a new miner's carry-in revenue

Action	Allocation Basis	Example Result
Settle 105 BEM from existing miners	Existing pool: cumulative consumption of 9 NAND and 1 LATCH	Participant share: 100 BEM; NAND 90, LATCH 10
Record the checkpoint	Preserve the cumulative entitlements above	Subsequent admissions do not rewrite them
Receive 105 BEM of carry-in revenue from the new miner	New miner: consumption of 1 NAND and 9 LATCH	Participant share: 100 BEM; NAND 10, LATCH 90
Begin the new period	Combined material ratio of 10 : 10	Subsequent ordinary revenue is allocated between classes at 1 : 1

Both amounts of 105 BEM above are gross revenue before the top-level revenue allocation; the participant amounts in the table are net of the other shares. After processing, the two participant classes each have a cumulative entitlement of 100 BEM, roles A and B each have 2.5 BEM, and C has 5 BEM. A new miner's “carry-in revenue” includes only the amount still unclaimed at admission and actually paid into the pool.

Provided that each miner's actual carry-in payment is unchanged across the admission sequences being compared, admitting miners individually, in batches, or in a different order within a batch does not change the class attribution of their carry-in revenue. If the external Mining protocol pays different amounts because of timing or remaining supply, this property does not establish that final revenue is independent of transaction order.

Before admission, anyone may trigger Mining's claim, which pays the miner's holder at that time. An amount settled to the previous holder before admission does not enter the pool again.

Holders do not need to claim before a new miner is admitted to protect their cumulative share from the previous period. Whether payment has already been withdrawn does not alter the entitlements preserved at the checkpoint. Ordinary revenue generated after the new miner enters the pool is allocated using the new pool-wide material composition.

6.4 The Fractional Ledger

The internal accounting scale is $\Lambda=84\times10^{18}$; one ledger unit equals $1/\Lambda$ of one BEM smallest unit. For x BEM smallest units, the exact participant share is recorded as $u(x)=x\times80\times10^{18}$ ledger units. Each class allocation rounds the NAND amount down and assigns the remainder to LATCH.

$$\begin{aligned}
&\Lambda = 84\times10^{18}, \qquad u(x) = 80x\times10^{18}\\
&d = u(T-T_0)\\
&U_N = B_N + \left\lfloor \tfrac{dN}{N+L} \right\rfloor\\
&U_L = B_L + d - \left\lfloor \tfrac{dN}{N+L} \right\rfloor\\
&I_N = \left\lfloor \tfrac{U_N}{\Lambda} \right\rfloor, \qquad I_L = \left\lfloor \tfrac{U_L}{\Lambda} \right\rfloor
\end{aligned} \tag{4}$$

The period formulas apply when N+L>0. T_0 is cumulative revenue at the previous boundary; B_N and B_L are the class bases, with $B_N+B_L=u(T_0)$. This allocation is not applied before the first admission, when no material composition exists.

$$\begin{aligned}
&v_{N,j} = \left\lfloor \frac{u(r_j)n_j}{n_j+l_j} \right\rfloor\\
&v_{L,j} = u(r_j) - v_{N,j}
\end{aligned} \tag{5}$$

r_j is the new miner's actual carry-in revenue, in BEM smallest units; n_j and l_j are that miner's own material consumption.

The checkpoint first updates B_N/B_L to the current U_N/U_L and sets T_0 to T. The carry-in amounts for each class are then added to the bases, T and T_0 both increase by r_j, and the pool-wide material composition is updated last. This preserves the historical ledger and avoids incorrectly applying either the old or the new pool-wide ratio to carry-in revenue.

6.5 Rounding and Invariants

$$\begin{aligned}
&\text{residue} = T - A - B - C - S_N q_N - S_L q_L\\
&\text{residue} \le 3 + 1 + \max(S_N - 1, 0) + \max(S_L - 1, 0)
\end{aligned} \tag{6}$$

Top-level rounding leaves at most 3 BEM smallest units, and class rounding at most 1; per-receipt rounding leaves at most S-1 smallest units for each nonempty class.

Rounding residue remains in the pool; no separate recipient account is created for it. Its value is recomputed from the cumulative ledger, so residues observed at different times must not be added together. Class entitlements and per-receipt amounts are governed by the integer implementation; the proportional values in the examples illustrate the allocation relationships.

$U_N+U_L=u(T)$, and I_N+I_L equals either P(T) or P(T)-1. Within a period of unchanged material composition, harvesting in several installments or in a single transaction produces the same cumulative result. Cumulative entitlements do not decrease; the current unclaimed balance decreases when a claim is paid.

6.6 New Miners and Differences in Revenue Within a Pool

Each pool allocates ordinary revenue according to the cumulative material proportions of all its admitted miners; it does not continuously attribute revenue to each individual miner. Differences in miner output are therefore shared within the pool. Differences in efficiency, optimal slot status, or failure can affect the relative revenue of the two classes.

Table 9. Hypothetical effect of a new miner on future class allocations

Hypothetical Scenario	Before Admission	After Admission
Cumulative pool material consumption, N : L	90 : 10	100 : 100
Participant revenue per period	100 BEM	110 BEM
NAND class allocation per period	90 BEM	55 BEM
LATCH class allocation per period	10 BEM	55 BEM

This example assumes that the new miner consumes 10 NAND and 90 LATCH and adds 10 BEM of participant revenue per period, while the existing miners' revenue remains unchanged. Total revenue increases, but the NAND class's future allocation decreases. No additional entitlement receipts are issued, and the historical ledger is not recalculated. This is an abstract illustration of the formula, not an actual miner design or a revenue forecast.

Admitting new miners neither issues additional entitlement receipts for this round nor rewrites previously accumulated entitlements. It can change the class proportions applied to future revenue, so future revenue for either class may increase or decrease.

A revoked miner stops generating revenue but remains in its original custody pool. Its historical NAND and LATCH consumption remains part of the pool's material composition. The contract does not automatically remove that consumption, so revoked miners continue to influence the class proportions of subsequent revenue.

The fixed allowlist can continue admitting eligible miners after ownership has been renounced. Locking the rules at deployment therefore does not mean that the future miner portfolio or class revenue proportions remain unchanged forever. [1,6,7]

7 Claims and Transfers

7.1 Claim History by ID

$$q_N(p) = \left\lfloor \frac{I_N(p)}{S_N} \right\rfloor, \quad q_L(p) = \left\lfloor \frac{I_L(p)}{S_L} \right\rfloor$$
$$\text{claimable}(p, c, i) = q_c(p) - \text{claimed}_p[c][i] \tag{7}$$

A class with zero supply has no entitlement receipts that can be claimed against, and no division by zero is performed. A pool cannot be activated if both class supplies are zero.

claim(category,id) and claimBatch(nandIds,latchIds) pay the caller and require the caller to currently hold every listed entitlement receipt. After a successful claim, the corresponding claimed value is updated to the current q. Claim histories are maintained independently by pool, class, and ID; a transfer does not reset them. Batch calls are subject to gas limits, and claims must be made separately in the two pools.

ERC-1155 operator approval does not itself grant permission to claim BEM. An approved operator may, however, transfer entitlement receipts under that approval, so holders must still manage their approvals carefully. The distributor neither claims BEM on behalf of beneficiaries nor redirects their revenue to third parties.

7.2 Transfers and Unpaid Entitlements

Suppose an entitlement receipt has 3 BEM unclaimed in the Optimal Pool and 7 BEM unclaimed in the Ordinary Pool. Its original holder claims the 3 BEM from the Optimal Pool and then transfers the receipt. The buyer can claim the remaining 7 BEM from the Ordinary Pool and subsequent additional entitlements, but cannot claim the 3 BEM already paid.

A seller may still claim revenue before a trade settles. Buyers should inspect the latest records for the specific ID in both pools. If the agreed price includes a particular unclaimed amount, the trading mechanism must enforce that condition; merely displaying a number does not lock it in.

After transfer, the former holder can no longer claim unpaid entitlements attached to that ID. Buyers must inspect the ID-specific records in both pools and should not assume that the amount unclaimed at listing will remain unclaimed until settlement.

7.3 Exit

The current design permits transfers of entitlement receipts but guarantees no minimum price, buyer, market making, or liquidity. Receipts cannot be redeemed for NAND, LATCH, or miners. Production pools have no miner withdrawal, migration, or liquidation exit. If revenue ceases, participants cannot redeem the miners to realize their residual value.

The right to transfer an entitlement receipt is distinct from the right to redeem the original materials. The current production design provides no redemption path for materials, miners, or liquidation value. Any future exit arrangement would need to be defined separately. [1,7,8]

8 Deployment Configuration and Authority

8.1 Contract Responsibilities and Deployment Parameters

Table 10. Contract instances and responsibilities for one processor

Contract or Library	Count	Responsibilities
TapeoutEntitlementReceipt	2	Issue NAND and LATCH entitlement receipts separately; maintain unique IDs and a holder index.
TapeoutReceiptDistributor	1	Hold receipt inventory, fix beneficiaries and ID ranges, and deliver entitlement receipts.
TapeoutMinerPool	2	Serve as the Optimal Pool and Ordinary Pool, respectively; custody miners, recognize revenue, and calculate and pay entitlements.
PoolMath	Library	Provide pure accounting calculations, with a corresponding implementation in the SDK.

Deploying both processors requires ten stateful instances in total. The two pools for each processor read the same pair of entitlement receipt contracts and are associated with the same distributor, but maintain independent miner registries, revenue ledgers, and claim records.

At construction, each pool fixes the receipt pair, BEM, Mining, miner collection, name, initial A/B recipients, and an allowlist of 1 to 16 depositor addresses. The allowlist has no modification entry point and survives ownership renunciation. The 80:1:1:2 weights and sink address C are code constants.

A pool requires the receipts' runtime code hashes to match the expected implementation and their reported classes to be correct. At construction, the distributor fixes two distinct pools with deployed code and checks that they report the same receipts, BEM, Mining, and miner collection. There is no registration entry point for expanding the pool list later.

Each pool can bind a distributor only once. The distributor must recognize that pool and report the same receipt pair. However, the two pools do not automatically read each other's bindings, and matching interfaces do not establish the distributor's code identity. A complete deployment manifest, verification of the actual code, and a cross-check that both pools bind the same distributor are therefore still required.

The website's source snapshot specifies Solidity 0.8.24, EVM paris, 200 optimizer runs, via-IR, and OpenZeppelin Contracts 5.4.0. Verification of an actual deployment must use the exact source and build parameters and address the tooling and compiler matters that the review identifies as still requiring verification.

8.2 Initialization Sequence

Table 11. Initialization and commencement of custody

Step	Expected Operation
1	Deploy the two receipt classes, two pools, and one distributor; confirm that processor and external protocol addresses are consistent.
2	Issue entitlement receipts and import the reconciled allocation plan; reconcile inventory, supplies, and all beneficiaries.
3	Finalize issuance for both classes, then finalize the allocation.
4	Bind both pools to the same correct distributor and activate them using the final supplies; verify code identity and the complete topology.
5	Renounce ownership of both receipt contracts, the distributor, and each pool.
6	Use the atomic admission entry point to admit miners to the intended pool through allowlisted depositors.

At a pool's first admission, issuance must be finalized for both receipt classes, supplies must equal S_N/S_L, and ownership of the receipt contracts must be renounced. The bound distributor's allocation must be finalized and its ownership renounced. The admitting pool must be activated and its ownership renounced. Because the receipts and distributor are shared, their conditions constrain both pools; each pool's own ownership condition constrains only that pool.

8.3 Operations Remaining After Finalization

Table 12. Operational authority after finalization

Actor After Finalization	Operations Still Permitted
Entitlement receipt holders	Transfer receipts and claim BEM for their receipts in each pool.
Anyone	Deliver receipts to their fixed destinations; harvest miner revenue and call sync; trigger A/B and C fee payments.
Fixed allowlisted depositors	Continue admitting eligible miners, changing the future miner portfolio and material composition.
Current A/B recipients	Transfer only their own recipient roles, without changing the fee rate.

Renouncing ownership restricts subsequent administrative actions but does not establish that the earlier allocation, miner selection, or deployment topology was correct. Because renunciation is irreversible, configuration errors may also be preserved permanently.

Both pools share the receipt contracts and distributor, so incomplete finalization or unrenounced ownership of those shared contracts constrains both pools. Each pool's own ownership and activation states are checked separately, and one pool may open after the other. A/B recipient roles and the fixed allowlist are independent of initialization ownership and remain in place after ownership is renounced. [1,4,5,7,8]

9 Security Design and Assumptions

9.1 Issuance and allocation constraints

After issuance is finalized, the number of entitlement receipts can no longer change, fixing the denominator for each class. After allocation is finalized, beneficiaries and their assigned ID ranges cannot be replaced. Fixed fee weights prevent administrators from subsequently changing the revenue split. These constraints depend on correct initial inputs; an inventory coverage check cannot replace reconciliation of the entire contribution round.

9.2 Admission and asset-transfer constraints

The supported admission procedure performs validation, settlement of existing miners' revenue, checkpointing, transfer of new miners, and attribution of carry-in revenue in one transaction. Failure at any step reverts the entire transaction. Initial admission also requires the relevant initialization authorities to have been relinquished. Production pools expose no interfaces for miner withdrawal, upgrades, migration, or arbitrary calls, restricting the authority to move admitted assets out later. Correspondingly, there is no general recovery mechanism for mistaken transfers, incorrect pool assignments, or changes in external dependencies.

9.3 Accounting and payment constraints

Revenue is recognized from actual increases in the pool's token balance; estimated revenue is not booked in advance. Accrued entitlements from the preceding period are preserved before the material composition changes. Each new miner's carry-in revenue is attributed using its own material consumption ratio, avoiding a recalculation of earlier revenue using the updated aggregate composition. Amounts already paid are recorded by pool, class, and receipt ID, so transferring a receipt does not allow the same entitlement to be claimed again.

9.4 Scope of verification

These constraints can be checked against the source code, state interfaces, and events of the corresponding version. They do not establish the correctness of off-chain eligibility decisions, miner selection, or initial allocations. Actual deployments also require verification of code identity and constructor parameters; values reported by contract interfaces alone are insufficient. Reconciliation and deployment verification before finalization, together with contract execution after finalization, form the system's controls. [1,5-9]

10 Release Status and Review Scope

This section records the status stated in the official website's source snapshot, labeled November 1, 2026, and its accompanying materials. It makes no claim about deployment progress after that snapshot. The candidate implementation, external assessment, and internal reviews each apply to their explicitly identified versions.

Table 13. Reference implementation and public release status

Item	Status disclosed in the source materials
Whitepaper and source snapshot	The original website text is labeled v1.0, with a source snapshot dated November 1, 2026.
Candidate implementation	TapeoutMinerPool/2.3.0, package 2.3.0-rc.1; frozen implementation commit 02ef1f7. The later documentation snapshot, 8819a59, contains the same contract source code.
Mainnet deployment	The source snapshot explicitly states that this candidate has not been deployed to mainnet. The inspected versions of the accompanying website pages contain no configured production contract addresses.
Contributions and claims	The historical contribution round has ended. The inspected materials provide no final reconciliation, issuance, or allocation results; web-based claims are not yet available.
Independent external assessment	The SALUS report is dated October 28, 2026 and covers the earlier c3cb933 / 2.0.0 version, not the current 2.3.0 candidate.
Current internal reviews	Version 2.3.0 has undergone project-internal reviews, including reviews using AI models. These are not independent third-party audits or guarantees of security.

The earlier SALUS report listed on the website contains three medium-severity, three low-severity, and two informational findings, all marked Pending in the original report. The implementation team has

reported code changes addressing some findings and has also recorded acceptance of certain operational risks. Neither substitutes for confirmation from SALUS that the findings have been closed.

The current internal reviews cover the dual-pool lifecycle and related tests. They also identify matters requiring verification, including the completeness of the deployment inventory, code identity, allocation coverage, tooling, and compiler configuration. Severity assessments differ across the internal reviews; citing only a statement that no high-severity issue was found would not represent their full conclusions.

An independent audit should identify its exact source commit and scope. An actual deployment additionally requires checks of addresses, bytecode, and initialization state. Published source code, passing tests, and owner renunciation each provide evidence, but none alone establishes that the system is secure under all conditions.

A static document does not automatically incorporate subsequent changes. Production status should be established from updated announcements, actual deployment addresses, verified code, and on-chain transactions. [1,4,9]

11 Operational Limits and Risks

11.1 External protocols and token behavior

The pools depend on BEM, Mining, and the miner collections. Mining weights, optimal slots, successful challenges, and the token issuance cap may all change output. A prior project investigation observed at a specific block that ownership of the Mining proxy had been renounced. The miner collections, however, use beacon proxies whose controllers can still change their implementations. Production-pool admission does not reverify the identity of external implementations; these dependencies must be checked at a specified block before production operation.

BEM is assumed to behave as a conventional ERC-20 token without transfer taxes, rebasing, or forced debits. Unexpected balance changes during payment cause the transaction to revert. If the balance falls below the amount the ledger requires the pool to retain, synchronization and claims may be blocked until the balance is restored.

11.2 Continued operation and gas limits

Measurements on a fork of the actual BSC network recorded in the source materials cover up to 24 miners, with an incremental admission-settlement cost below 28,000 gas for each additional existing miner. The 400-miner scale was measured using mocks. Extrapolation from the then-applicable per-transaction limit of 2^{24} gas produced an estimated capacity of approximately 580 to 590 miners; the tooling's planning threshold was 401.

These are historical tests and estimates, not guaranteed safe capacities or permanent network parameters. The two pools have separate registries and capacity limits. Before production operation, the intended miner mix should be tested against the chain limits then in effect, with sufficient headroom retained.

Each new admission settles all existing miners, so its cost grows linearly with the number registered. Except for REVOKED miners, a failed claim from any existing miner can revert a new admission. After admission capacity is reached, revenue collection in batches and existing entitlement claims can continue, but this does not guarantee availability under every external failure condition.

revokeWallet revokes all allocations for one wallet in a single transaction. An excessively large erroneous allocation can make this transaction exceed the gas limit, preventing correction even before finalization. Receipt delivery and batch claims must likewise be divided into batches that fit the available capacity.

11.3 Initial custody, permanent custody, and market risk

Between the transfer of materials and completion of receipt allocation, participants depend on the operator to handle the materials and records. Omitted transfers, incorrect classes, incorrect beneficiary addresses, or incompatible wallets may cause misallocations or make delivery permanently impossible. Finalization and owner renunciation lock in the preceding configuration; they do not establish or correct its accuracy.

Once admitted, a miner cannot be moved to another pool or withdrawn, and revoked miners remain in the material composition. Future revenue may fall or reach zero. Neither the execution price nor the ability to sell an entitlement receipt is guaranteed. Loss of control over an A/B recipient address may render that role's share inaccessible without changing the allocation ratios of other accounts.

The contracts promise no principal protection, minimum payment, rate of return, exchange rate, or fiat value. They define how actual revenue is allocated, not how much revenue the external protocol will produce. [1,5-10]

12 Public Verification and Further Work

Public verification requires contribution records, receipt allocations, miner inventories, and BEM ledgers to be reconciled with one another. The contribution wallet's current balance is not the cumulative quantity of eligible contributions; expected receipts are not issued assets; and estimated pending Mining rewards are not revenue already received by a pool.

Table 14. Verification evidence required for production operation

Verification item	Evidence to provide
Code and deployment identity	The complete inventory of five instances per processor, chain ID, creation transactions, source commit, build parameters, verified source code, and runtime code.
Bindings of the two pools	Both pools are bound to the same correct distributor; the distributor's fixed pool addresses match the deployment inventory. Interface reports do not replace code verification.
Round-wide allocation	Eligible material records, final supply by class, inventories, and allocations of every ID agree, with no omissions, duplication, or excess issuance.
Recipient compatibility and corrections	Beneficiary wallets can receive batch ERC-1155 transfers; errors can be corrected before finalization; policies for exceptional transfers and unused materials are public.
Activation and permissions	Issuance and allocation are finalized, denominators agree, relevant ownership is renounced, and the fixed allowlist and A/B/C configuration are correct.

Table 14. Verification evidence required for production operation

Verification item	Evidence to provide
Miners and revenue	Per-pool miner inventories, consumption and status, atomic admission transactions, ledgers and actual balances, per-ID claims, and fee-payment records.
External protocols and capacity	Upstream implementations and permissions at a specified block, gas measurements for the actual miner mix, failure-handling procedures, and conditions for stopping new admissions.
User interfaces and review	The website connects to the actual deployment; delivery and claims have been verified; an independent audit identifies its version, scope, and unresolved findings.

The website should distinguish undeployed contracts, inactive pools, unavailable data, stale data, and genuine zero values. It may aggregate entitlements across both pools for a processor, while still showing the underlying claim records separately. The claim buttons in the source materials are disabled placeholders; the existence of a contract function does not mean the website has enabled the corresponding operation.

Operational matters such as material selection, miner assignment, public reconciliation and corrections, exceptional transfers, unused materials, losses, and responsibility for expenses should be specified in separate, traceable public policies. Each item in the complete verification checklist should have supporting evidence. An incomplete checklist cannot establish that the entire system has been verified.

Further work may include interfaces for receipt delivery and claims, cross-pool entitlement summaries, exportable reconciliation records, and monitoring of miner status and upstream implementations. Per-miner revenue attribution, removal of revoked miners' consumption, refunds, or migration would change the economic rules. Such changes require separate design and an assessment of their effects on existing participants' entitlements. [1,3-5,9]

References

[1] TapeMine. Technical Whitepaper v1.0. The official text labels its source snapshot 1 November 2026. tapemine.com/whitepaper.

[2] TapeMine. Official website and contribution-round status. tapemine.com/.

[3] TapeMine. Pool rules and allocation policy. tapemine.com/pool-rules.

[4] TapeMine. Contract architecture and build information. tapemine.com/contracts.

[5] TapeMine. ARCHITECTURE.md. Pinned implementation commit 02ef1f7. GitHub: docs/ARCHITECTURE.md.

[6] TapeMine. ACCOUNTING.md and PoolMath.sol. Pinned implementation commit 02ef1f7. GitHub: docs/ACCOUNTING.md; GitHub: src/PoolMath.sol.

[7] TapeMine. TapeoutMinerPool.sol. Pinned implementation commit 02ef1f7. GitHub: src/TapeoutMinerPool.sol.

[8] TapeMine. TapeoutEntitlementReceipt.sol and TapeoutReceiptDistributor.sol. Pinned implementation commit 02ef1f7. GitHub: src/TapeoutEntitlementReceipt.sol; GitHub: src/TapeoutReceiptDistributor.sol.

[9] TapeMine. External assessments, internal reviews, and risk-acceptance records. tapemine.com/audit.

[10] TapeMine. MINING_LIFECYCLE.md. Pinned implementation commit 02ef1f7. GitHub: docs/MINING_LIFECYCLE.md.