Structure Overview Key Points

- The airdrop covers **34% of total FINT supply (170 million tokens)**.
- It is divided into **three phases (Genesis, Season 1, and Season 2)** spread over **9 months**.
- Each phase uses the same **points-based and pro-rata distribution model**.
- The goal is to reward real user activity (trading, liquidity, referrals) and maintain engagement beyond launch.

Phases

1. **Genesis Phase (Months 0–3):** Starts at mainnet launch. Rewards early users for on-chain activity like trading and referrals to bootstrap liquidity.
2. **Season 1 (Months 3–6):** Distributes 12% of total supply. Focuses on trade mining and active user participation.
3. **Season 2 (Months 6–9):** Distributes another 12% of supply. Introduces new platform features or goals to sustain activity.

Each phase ends with token distribution proportional to user points earned during that period.

Points and Snapshots

- Points are assigned based on four activity categories:
    1. **Trading Volume** – converted to points.
    2. **Maker Orders** – rewards for adding liquidity.
    3. **Trade Frequency** – rewards for consistent participation.
    4. **Referrals** – rewards for bringing active new users.

- **Snapshots** are taken weekly or bi-weekly during each season.
- At every snapshot:
  `F_total = F1 + F2 + F3 + F4`
- Across a season:
  `F_season = sum of all snapshot scores for that user`

Distribution Calculation

At the end of each season, tokens are distributed using this formula:

`user_tokens = (user_F_season / total_F_season_all_users) × pool_tokens`

Each user's tokens are proportional to their total points compared with all participants.

Eligibility and Anti-Gaming

- Minimum activity thresholds are required to prevent spam or fake trades.
- Referral points activate only when the referred user becomes active.
- Wash trading or circular trades receive zero points.
- On-chain clustering identifies and limits sybil behavior.
- Scoring parameters can be updated for future snapshots but not retroactively.

Implementation Notes

- Weekly snapshot tasks can be handled by off-chain workers or runtime scheduler.
- Each season has a defined reward pool (e.g., 45M FINT).
- Snapshots and points are stored on-chain as mappings per user.

- Claiming is activity-gated, meaning users must reach certain engagement levels before being eligible to claim.
- Unclaimed or invalid tokens after each phase are burned or returned to treasury.

Developer Parameters for Genesis Configuration

- **Total airdrop supply:** 170,000,000 FINT (28% of total).
- **Snapshot interval:** 7–14 days.
- **Season duration:** 3 months.
- **Point weights:** adjustable per activity type.
- **Claim window:** ~30 days after season end.
- **Unused tokens:** burned or recycled to treasury.
- **Distribution model:** pro-rata based on cumulative points.

Collator Staking and Inflation Model Key Points

Fintradex follows the **Polkadot-style staking and inflation model** where collators and delegators secure the parachain through delegated staking. The system is designed to balance **security, decentralization, and liquidity** by adjusting rewards dynamically based on staking participation.

The **Relay Chain** (Paseo/Polkadot) provides finality and shared security, while the **Fintradex parachain** handles execution, state updates, and internal staking logic for collators.

Collator Staking Mechanism

- **Collators** maintain parachain state, build blocks, and submit them to the Relay Chain with proofs of validity.

- **Delegators** stake their tokens by nominating collators and earn a share of rewards based on the amount delegated.
- Collators are chosen for each session based on total backing (their own stake plus delegations) and uptime performance.
- Rewards are distributed each round from the **staking reward pool**, split between collator commission and delegator payouts.
- To prevent manipulation, only the top delegations per collator are eligible for reward distribution.
- Collator exits are subject to an **unbonding delay**, and slashing applies for downtime or invalid block submissions.

Inflation and Reward Distribution

- The network mints a fixed annual issuance of $FINT that acts as the **security budget**.
- This issuance is divided between **stakers (collators + delegators)** and the **treasury** based on real-time staking participation.
- If staking participation is below the target (network under-staked), the reward rate automatically **increases** to attract more delegation.
- If participation exceeds the target (over-staked), the reward rate **decreases**, and the surplus is redirected to the treasury.
- This creates a **self-balancing system** that keeps staking yield stable and avoids excessive token lock-up.

Formula Concept (Simplified)

Let:

- s = current staking ratio (total staked / circulating supply)

- `s*` = ideal staking target
- `I` = annual token issuance
- `f` = minimum base reward rate

Then:

- When `s < s*`, staking rewards increase proportionally to attract more participants.
- When `s > s*`, staking rewards decrease, and excess tokens flow to the treasury.
- The overall goal is to keep `s` near `s*` while maintaining predictable annual issuance.

Roles and Reward Flow

- **Collators:** Earn commission plus proportional block production rewards.
- **Delegators:** Earn rewards pro-rata from their chosen collator's staking pool.
- **Treasury:** Receives any unallocated inflation and can fund grants, development, or buybacks.
- **Governance:** Adjusts key parameters such as ideal staking ratio, inflation rate, and delegation limits through on-chain proposals.

Economic Rationale

- The model ensures **network security** by rewarding staking when participation is low and capping rewards when the network is saturated.
- It maintains **predictable inflation** but flexible distribution based on live staking conditions.
- It avoids **over-rewarding locked capital** and instead redirects excess emissions to the treasury for productive use.

- The adaptive algorithm creates a sustainable balance between token liquidity and network security.

Developer Parameters for Genesis Configuration

- **Initial annual issuance (I):** fixed total to be defined (e.g., 5–10% of supply).
- *Target staking ratio (s):** ideal percentage of total supply staked (e.g., 50–60%).
- **Minimum floor rate (f):** baseline yield to ensure constant reward availability.
- **Commission range:** configurable per collator (e.g., 5–10%).
- **Unbonding period:** defined in blocks or eras (e.g., 7–14 days).
- **Reward distribution cadence:** per era or per block, adjustable by governance.

Example (Simplified Numbers)

- Total annual issuance (I) = 5% of total supply.
- Target staking ratio (S*) = 50%.
- Current staking ratio (S) = 40%.
- Floor rate (f) = 2%.

Since S < S*, rewards increase.
Extra = (S* - S) / S* = (50 - 40) / 50 = 0.2
R(pool) = I × (1 + Extra)
R(pool) = 5% × 1.2 = 6%

If S = 70% (> target):

Extra = (S - S*) / S* = (70 - 50) / 50 = 0.4
R(pool) = I × (1 - Extra)
R(pool) = 5% × 0.6 = 3%

veTokenomics: Time-Locked Voting and Boosted Rewards Key Points

Core Concept

- The `veFINT` (vote-escrowed FINT) system converts standard $FINT into non-transferable governance and reward tokens by **locking them for a fixed period**.
- This design links **governance power + fee rewards + liquidity incentives** directly to the *duration* of commitment, not just token quantity.
- It enforces long-term alignment between token holders and protocol growth.

Locking Mechanism

Users lock `FINT` for up to 4 years and receive `veFINT` based on a linear time-weight formula:

```
veFINT = FINT × (lock_time_remaining / max_lock_duration)
```

-
    - 4-year lock → 1.0 × FINT = 1 : 1 veFINT
    - 3-year lock → 0.75 × FINT
    - 2-year lock → 0.50 × FINT
    - 1-year lock → 0.25 × FINT
- `veFINT` **decays linearly** until unlocked; users can extend the lock or add more FINT to refresh their weight.
- Only **one active lock** per address is allowed, simplifying on-chain accounting.

Governance & Security Logic

- **Time-weighted voting** ensures that power comes from commitment, not temporary speculation.
- Prevents short-term actors from buying, voting, and dumping.

- Decay + single-lock structure forces continual engagement to retain influence.
- Ideal for on-chain governance pallets (`pallet-referenda`) because each block can recompute veFINT weight from stored lock expiry.

Reward Integration

- `veFINT` holders get **boosted staking yields** and **fee-sharing rights**.
- Boost factor scales with `lock_time_remaining / max_lock_duration`.
- Rewards decline naturally as veFINT decays → requires periodic relocking to maintain full yield.
- Rewards are sourced from **real protocol trading fees (USDC, ETH, etc.)**, *not inflation*, ensuring non-dilutive "real-yield" returns.

Implementation Notes for Chain Runtime

- **Storage:**
  - `Locks::<AccountId, LockInfo { amount, unlock_block }>`
  - `veBalance::<AccountId, Balance>` (computed as function of time_remaining).
- **Hooks:**
  - On every `on_finalize`, update decay and eligible rewards.
  - Integrate into `pallet-rewards` or `pallet-treasury` for fee distribution.
- **Governance Integration:**
  - Replace simple `balance_of` voting weight with `veBalance`.
- **Decay Calculation:**

- Linear decay per block = `veFINT_initial / lock_duration`.

Economic Outcome

- Encourages **long-term staking**, **stable governance**, and **supply reduction** (soft sink).
- Creates continuous demand for relocking to maintain voting and reward power.
- Distributes **real revenue** proportionally to committed users, enhancing protocol sustainability.

Adaptive Buyback with Emergency Fund Key Points

Core Concept

- Fintra introduces an **Adaptive Supply Pool (ASP)** — an automated, rule-based system that uses **protocol-generated revenue** to:
    1. **Buy back and burn $FINT tokens**, reducing circulating supply when the system is healthy.
    2. **Rebuild the Emergency Fund**, providing liquidity and reserves during adverse or volatile conditions.
- The mechanism operates entirely on-chain, without human discretion, ensuring **transparency and predictable monetary policy**.

Mechanism Overview

- Each epoch, a fixed share of **net trading fees** (after discounts and rebates) flows into the **ASP**.
- The ASP automatically routes funds between:
    - **Buyback Pool:** Purchases $FINT from open markets → burns it on-chain.

    - **Emergency Fund (Reserve):** Accumulates stable assets (e.g., USDC) for liquidity backstops, operational continuity, and risk mitigation.

- This adaptive controller ensures **counter-cyclical behavior**:
    - In growth phases → prioritize **buybacks** to compress supply.
    - In stress phases → prioritize **reserve accumulation** to strengthen resilience.

Rule-Based Routing Logic

- **Safety Checks (Priority 1):**
    - If the reserve balance < minimum threshold (floor)
    - If fee inflows drop or market depth is thin
      → Divert all ASP inflows to the Emergency Fund until stability is restored.
- **Reserve Bands (Priority 2):**
    - **Below floor:** 100% → Reserve
    - **Between floor & ceiling:** Dynamic split → Reserve ↔ Buyback curve
    - **Above ceiling:** 100% → Buyback
- **Non-zero floors:** Even during extremes, small fixed percentages always flow to both sides to ensure continuous accumulation and deflation.

Economic Flow Summary

1. **Protocol Fees → Adaptive Supply Pool (ASP)**
2. **ASP splits → [Buyback Pool | Emergency Fund]**
3. **Buyback Pool → Market purchase of $FINT → Burn**
4. **Emergency Fund → Reserve wallet (USDC/Stablecoins)**

Formulaic Representation:

```
ASP_split = f(circulating_supply, reserve_ratio, market_health)
Buyback_share = curve(reserve_ratio, supply_band)
Reserve_share = 1 - Buyback_share
```

Where:

- `reserve_ratio` = Current_reserve / Target_reserve
- `supply_band` = [lower_bound, upper_bound] thresholds defining burn vs reserve bias.

Implementation Notes for Runtime / Treasury Pallet

- **Storage:**
    - `ReserveBalance` (USDC/Stable equivalent)
    - `BuybackBudget` (FINT burnable pool)
    - `SupplyBands { lower, upper }` for curve thresholds.
- **Hooks:**
    - Trigger on `on_finalize` every epoch to compute ASP routing.
    - On surplus → call `burn_fint()` extrinsic.
    - On deficit → route to `reserve_account`.
- **Governance Parameters:**
    - Adjustable via OpenGov proposals:
        - `floor_ratio`, `ceiling_ratio`
        - `min_buyback_rate`

        - `emergency_trigger_conditions`

Economic Outcomes

- **Deflationary Pressure:** Buybacks systematically offset emissions, maintaining token scarcity.
- **Resilience:** Emergency Fund ensures operational continuity and liquidity backstops during market downturns.
- **Transparency:** All routing decisions are on-chain and verifiable through published parameters.
- **Counter-Cyclical Stability:**
    - Bull markets → aggressive buybacks → price support.
    - Bear markets → reserve accumulation → runway protection.

Supply-Side Economy Key Points

Core Principle

- Total supply is **hard-capped at 500 million $FINT**.
- No inflation; all rewards come from **protocol fees** and **buyback burns**.
- Circulating supply evolves only through **scheduled vesting unlocks**, **airdrop releases**, and **burn events**.

Token Allocation Structure

1. **Public IDO — 20 % (100 M $FINT)**
    - **Delay:** None | **Vesting:** Immediate
    - **Release:** 20 % at TGE (Day 0)

- Provides early market liquidity and price discovery.

2. **Initial Airdrop (Genesis) — 10 % (50 M $FINT)**
    - **Delay:** None | **Vesting:** Immediate
    - **Release:** 10 % at TGE (Day 0)
    - Distributes ownership to early community participants.
3. **Liquidity Programs — 6 % (30 M $FINT)**
    - **Delay:** None | **Vesting:** Immediate
    - **Release:** 6 % at TGE (Day 0)
    - Rewards LPs and market-makers for depth and tight spreads.
4. **Airdrop Season 1 — 12 % (60 M $FINT)**
    - **Delay:** None | **Vesting:** Tranche unlock at Month 3
    - **Release:** 12 % distributed after points snapshot at Month 3.
5. **Airdrop Season 2 — 12 % (60 M $FINT)**
    - **Delay:** None | **Vesting:** Tranche unlock at Month 6
    - **Release:** 12 % distributed after points snapshot at Month 6.
6. **Advisors & Marketing — 5 % (25 M $FINT)**
    - **Delay:** 6 months | **Vesting:** 12 months linear
    - **Monthly Release:** 0.417 % of total supply (~2.08 M FINT/mo)
    - **Total Duration:** 18 months.
7. **Team & Founders — 10 % (50 M $FINT)**
    - **Delay:** 12 months cliff | **Vesting:** 36 months linear
    - **Monthly Release:** 0.278 % (~1.39 M FINT/mo)

    - **Total Duration:** 48 months.

8. **Development Pool — 10 % (50 M $FINT)**
    - **Delay:** 12 months | **Vesting:** 36 months linear
    - **Monthly Release:** 0.278 % (~1.39 M FINT/mo)
    - **Total Duration:** 48 months.
9. **Treasury Reserve — 15 % (75 M $FINT)**
    - **Delay:** 12 months | **Vesting:** 36 months linear
    - **Monthly Release:** 0.417 % (~2.08 M FINT/mo)
    - **Total Duration:** 48 months.

Vesting and Locking Logic

- Implement all time-locks via `pallet-vesting` or custom runtime hooks.
- Team/Dev/Treasury accounts start at `TGE + 12 months`; linear monthly releases for 36 months.
- Advisors start at `TGE + 6 months`; linear for 12 months.
- Airdrop seasons and IDO allocations minted to claim or distribution pallets at TGE.
- Any unclaimed airdrop tokens → **burned** (hard sink).
- Integrate **Adaptive Supply Pool (ASP)** to route protocol fees → buyback/burn or reserve.
- Treasury and Dev fund controlled via OpenGov referenda.

Economic Outcomes

- **Community at TGE:** Public IDO (20 %) + Genesis (10 %) + Liquidity (6 %) = **36 % circulating** (≈ 180 M $FINT).
- **Total Airdrops:** 10 % + 12 % + 12 % = **34 % (170 M $FINT)**.
- **Locked at TGE:** ≈ 64 % (Team, Treasury, Dev, Advisors etc.).
- **Emissionless Model:** No minting post-TGE; all rewards funded by real fee revenue.
- **Deflationary Pressure:** ASP burns offset supply growth and reduce circulating float.
- **Governance Transparency:** All vesting and burn events verifiable on-chain.