Monetizing the WalletConnect Network

Towards Sustainable Fees for Onchain Connectivity

WalletConnect

November 2025

1. Summary

Since its inception in 2018, WalletConnect has become a foundational layer of onchain infrastructure enabling seamless, secure, remote connections between wallets and apps. Over the years, the WalletConnect Network has grown to support more than 700 wallets and 80,000 apps, facilitating over 380 million secure connections and enabling billions of dollars in onchain value transfer. From mobile wallet interactions to institutional DeFi access, WalletConnect has powered critical use cases across the ecosystem.

In 2025, the WalletConnect Network transitioned to a decentralized architecture. Today, 21 independent Node Operators maintain the network, and the introduction of the WalletConnect Token (WCT) has enabled decentralized governance, staking, and the introduction of rewards programmes for ecosystem participants.

- Wallet Partners receive token rewards for driving user activity and adoption.
- Node Operators earn rewards for providing reliable connectivity and decentralizing the network.
- Stakers participants are rewarded for their long-term commitment to the network.

To date, these rewards have been funded from a pre-allocated "rewards pool" comprising [17.5]% of the initial token supply. However, this rewards pool will be depleted over time, meaning that maintaining this level of infrastructure and aligning incentives across all participants will not be sustainable without introducing fees. To date, none of the participants of the WalletConnect Network has paid to use its services.

This paper proposes a sustainable fee mechanism to address that imbalance. In the original WalletConnect Network whitepaper, we envisaged that apps, which typically derive substantial value from the Network, would eventually become fee-paying. Here, we introduce a proposal for application-side fees based on volume and yield, metrics that reflect the economic value flowing through the WalletConnect Network from wallets to apps.

As a first step, it proposes a relatively simple monetization framework, designed to minimize implementation complexity and provide a clear path to adoption. It is not intended to represent the final or permanent model for network monetization. Rather, it serves as a starting point, one that can evolve through community input, ecosystem scrutiny, and iterative improvements.

Before implementation, this proposal will be subject to open discussion and a governance vote, ensuring alignment with the broader WalletConnect community and its long-term vision.

2. The Rationale Behind Application-Side Monetization

Applications (apps) in the onchain ecosystem offer valuable services to users, but critically, they do not operate in isolation. Every app-to-user interaction depends on a secure and trusted communication bridge between a wallet (where the user holds funds and signs transactions) and the application interface. Since 2018, WalletConnect has provided this bridge, enabling real-time, secure, and seamless connections between wallets and apps without requiring users to compromise on custody or experience.

While apps provide the front-end service, wallets and the WalletConnect Network jointly enable access to the user and to the value they hold. Wallets safeguard the user's private keys and interface for signing transactions. WalletConnect delivers the communication protocol that connects the two sides in a decentralized and interoperable manner. Without this infrastructure, value exchange between users and apps, especially across wallets and platforms, would be fragmented or entirely infeasible.

As the onchain economy matures, a large share of financial value flows through WalletConnect-powered connections. Apps are often the monetary beneficiaries of this value, whether through fees, spreads, revenue-generating actions, or broader business models.

To date, no fees have been paid for using the WalletConnect Network, creating a fundamental imbalance in the value chain.

Over the years, WalletConnect has facilitated billions of dollars in transaction volume and hundreds of millions of secure connections, directly powering the economic activity and growth of thousands of applications; yet none of this value has contributed back to sustaining the shared infrastructure that enables it. Introducing fees on the application side:

- Aligns incentives: Applications that benefit from WalletConnect-enabled transactions also contribute back to the infrastructure.
- Ensures sustainability: Wallet partners, Node operators, and governance participants can be adequately and fairly compensated.
- Preserves neutrality: Users remain unaffected by fees; they continue using their preferred wallets without changes to their experience.

This approach recognizes that value flows from users (via wallets) to apps, and that apps should contribute to the infrastructure that makes this value flow possible.

3. Objective

The primary objective of introducing WalletConnect fees is to ensure the economic sustainability and long-term growth of the network.

It seeks to align incentives among the core participants of the ecosystem: wallets, applications, and node operators, while maintaining a neutral, open, and reliable connectivity layer for the financial internet.

Applications contribute fees based on the financial activity they generate through the network.

End users will not pay to use WalletConnect. The experience remains smooth, free, and uninterrupted.

Fees apply only at the application layer, ensuring that wallets and their users continue to connect, transact, and interact as before.

WalletConnect fees apply only when the network facilitates value-bearing transactions.

Applications that use WalletConnect solely for authentication, wallet login, account linking, or other non-financial interactions do not incur fees.

This ensures a usage-based and value-aligned model, where applications contribute in proportion to the economic value they capture, rather than paying a blanket charge for network access.

In the following sections, we outline the two main categories used to measure financial activity and determine fees.

- Wallets receive rewards for enabling user access, safeguarding private keys, and executing onchain transactions on behalf of users.
- Node Operators earn rewards for running the WalletConnect infrastructure in a secure, decentralized, and regionally distributed manner, ensuring network reliability and performance.

3.1 Rewards Distribution

In this model, rewards are determined by the actual revenue generated by the network, ensuring that incentives scale directly with network performance.

Revenue Distribution Model

Let:

- R = Total Protocol Revenue,
- B_{w} = Base Wallet Rewards (USD-based, converted to WCT),
- B_{n} = Base Node Rewards (USD-based, converted to WCT).
- Surplus: $S = R - B_{\mathrm{w}} - B_{\mathrm{n}}$.

Revenue is allocated as follows:

1. Wallet Rewards

$$\text{Wallet Rewards} = B_{\mathrm{w}} + \phi_w\, S,$$

2. Node Rewards

$$\text{Node Rewards} = B_{\mathrm{n}} + \phi_n\, S,$$

3. Protocol Treasury

$$\text{Treasury Allocation} = \phi_t\, S,$$

Where: ϕ_w, ϕ_n, ϕ_t are governance-defined allocation parameters such that:

$$\phi_w + \phi_n + \phi_t = 1$$

The Protocol Treasury funds:

- Ongoing research and development,
- Strategic growth initiatives,
- Potential top-ups to the Flexible Rewards Vault.

The Flexible Rewards Vault (a fixed share of the total supply) subsidizes base rewards if:

$$R < (B_\mathrm{w} + B_\mathrm{n})$$

- In such cases, the vault covers the shortfall to maintain minimum incentives for wallets and node operators.
- Staking rewards are funded separately from this vault, as part of the protocol's monetary policy, rather than operational income.

3.1.1 Key Advantages

- **Performance-based** — Rewards grow in line with actual network revenue.
- **Currency-aligned** — USD-based calculation, WCT-based payment.
- **Built-in sustainability** — Treasury funding ensures long-term growth investment.
- **Shock resilience** — Flexible Rewards Vault maintains base incentives in downturns.
- **Governable** — Allocation parameters ϕ_w, ϕ_n, ϕ_t can be adjusted via governance, or even modeled as functions of revenue growth.

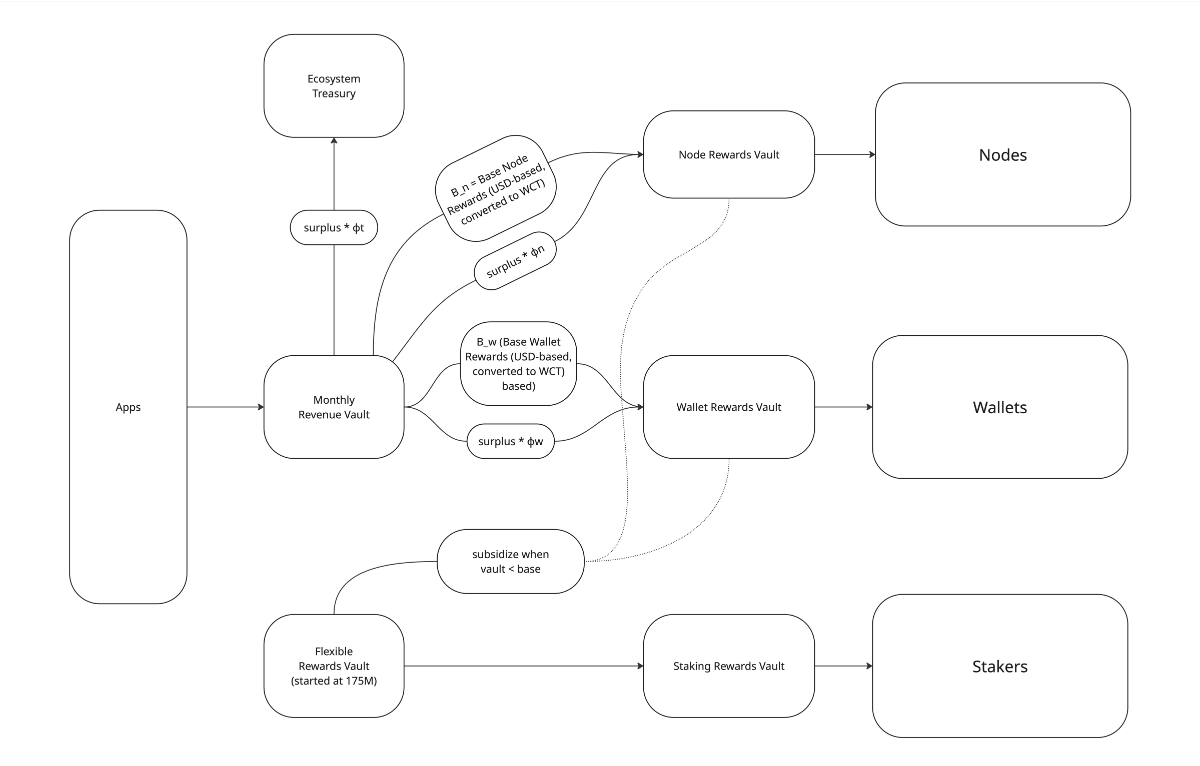

Figure 1: Protocol revenue and rewards distribution flow.

4. Categorization

To provide the fairest and most accurate pricing mechanism, the WalletConnect Network introduces a categorization framework that recognizes the different ways applications generate value through onchain financial activity.

The objective is to ensure that the network's monetization framework remains aligned with each application's underlying business model, so that fees are proportional to the real economic value derived from WalletConnect-facilitated activity.

Through analysis of network activity and common revenue models across the decentralized application landscape, WalletConnect identifies two primary categories of financial activity:

- (i) volume-based activity
- (ii) yield-based activity

Importantly, this categorization happens at the smart contract level, not at the application or project level. This means that each transaction is analyzed based on the specific contract calls executed within a wallet-signed transaction and is classified into one of the two categories accordingly.

This enables fine-grained billing accuracy — so if an application offers multiple types of financial services (e.g., swaps and staking), each component is billed according to its actual financial activity type.

Billing applications are based on their project identifier (project_id). This mechanism will be further reinforced through our Certification program, where, similar to how applications are required to verify their URL domain, projects will now also need to register the smart contracts they rely on. These contracts will be verified and then used as the basis for accurate categorization within the monetization framework.

4.1 Volume-Based Financial Activity

Definition:

Volume-based activity refers to economic interactions where value is generated per transaction or unit of volume processed.

This includes cases where a protocol captures a fee, spread, or commission from each transfer or swap facilitated through WalletConnect.

Examples of volume-based contracts include:

- Swap or trading contracts in DEXs and aggregators
- Bridge contracts processing cross-chain transfers
- Payment and checkout contracts facilitating token settlements

Characteristics:

- Value scales with transaction count or total notional volume.
- Financial events are discrete — each transaction generates a measurable instance of economic activity.
- The network's role is primarily execution and connectivity, enabling reliable message relays and transaction signing between wallets and applications.

Pricing Implication:

These activities are priced using the deterministic volume-based fee function, which calculates fees as a function of Total Network Volume (TNV).

This ensures that contracts handling greater throughput — and thus generating higher transactional revenue — contribute proportionally to network sustainability.

4.2 Yield-Based Financial Activity

Definition:

Yield-based activity refers to financial interactions where value is generated continuously over time, through yield accrual, staking, or interest-bearing mechanisms.

In these cases, WalletConnect facilitates not only deposits and withdrawals but also the ongoing management of time-dependent positions.

Examples of yield-based contracts include:

- Lending or borrowing contracts generating interest over time
- Staking and restaking contracts, distributing yield or rewards
- Liquidity vaults and structured-yield products

Characteristics:

- Revenue depends on time, rate, and balance, rather than transaction count.
- Value accrues continuously as yield or interest, realized only upon withdrawal or claim.
- The network's role extends to lifecycle management — enabling recurring user interactions across deposit, harvest, and claim events.

Pricing Implication:

These activities are priced using a yield-based function, which considers the realized yield on a position.

Fees accrue only when yield is withdrawn or realized, aligning network monetization with actual economic reward rather than nominal transaction flow.

4.3 Smart Contract–Level Categorization

Categorization is determined at the smart contract level, not the overall application level.

This distinction ensures granular billing accuracy, allowing hybrid applications — such as those combining swaps and yield products — to be billed proportionally to their respective financial activities.

For instance:

- Swap or Bridge contracts are categorized under the volume-based model, applying the volume-based fee function.
- Staking or Lending contracts are categorized under the yield-based model, applying the yield-based function.

This structure prevents overgeneralization and ensures that each portion of an app's ecosystem is billed according to its true economic purpose.

4.4 Certified App Program and Data Integrity

The WalletConnect Network leverages its Certified App Program to ensure high-quality data labelling and reliable categorization.

Through this program:

- Applications undergo contract verification and audit to confirm ownership, purpose, and security.

- Verified smart contracts are tagged according to their financial activity type (volume-based or yield-based).
- These mappings are maintained within WalletConnect's onchain and offchain registries, enabling precise classification at the transaction level.

This approach mirrors WalletConnect's historical verification of URL domain ownership — extending the same rigor to smart contract validation.

As part of the certification process:

- Each app's verified contract set must be declared and audited.
- Transactions routed through WalletConnect are automatically matched to their certified contract type.
- Multi-product applications (e.g., liquidity provision and swaps) are accurately disaggregated, ensuring each activity is billed under the correct fee model.

4.5 Evolution and Adaptability

As the onchain ecosystem evolves, new financial primitives and revenue models will emerge.

The WalletConnect Network is designed to adapt dynamically — introducing new pricing mechanisms or refining existing ones to reflect these innovations.

This adaptability ensures that the protocol remains fair, competitive, and sustainable, regardless of how decentralized finance and application models continue to evolve.

5. Volume-Based Fee Function

The volume-based fee function applies to applications where transactional volume represents the main source of value creation — such as trading, bridging, or payments.

This model ensures predictable, value-aligned monetization for the majority of the ecosystem.

Fees are determined by a continuous, non-linear function that dynamically adjusts based on each application's recent economic activity.

This structure ensures fairness, scalability, and predictability — applications with higher sustained usage contribute more in absolute terms while benefiting from progressively lower effective rates.

5.1 Understanding TNV

Total Network Volume (TNV) is the core metric used to determine fees for volume-based applications. It captures the total economic value transmitted through the WalletConnect Network, representing the aggregate token value exchanged in transactions initiated by wallets during WalletConnect sessions. Each transaction hash collected through the Wallet SDK is analyzed, and the corresponding token transfers are parsed and summed to calculate TNV. This provides a reliable proxy for the financial activity facilitated between wallets and applications.

5.1.1 How TNV is Calculated

For each transaction initiated via a WalletConnect session:

1. **Capture** — The Wallet SDK collects transaction hashes generated during WalletConnect sessions.

2. **Parse** — Each transaction is programmatically analyzed to identify all token transfers on supported blockchains.
3. **Evaluate** — The inflows and outflows related to the initiating wallet address are summed.
4. **Compute** — The TNV for a transaction is defined as the maximum of the total inflow or outflow for the wallet.

$$\text{TNV}(w) = \max(O(w),\, I(w))$$

where:

- w is the wallet address,
- $O(w)$ is the total outflow from w,
- $I(w)$ is the total inflow to w.

This ensures that the captured value reflects the most significant directional flow of assets while avoiding artificial inflation from internal contracts hops.

5.1.2 Transaction Classification

Table 1: Transaction classification for TNV computation

#	Type	Description	Example
1	Single-State Outflow	Wallet sends tokens	ETH or USDC transfer
2	Single-State Inflow	Wallet receives tokens	Airdrop
3	Multi-State Outflow	Wallet sends multiple assets	Batch token transfer
4	Multi-State Inflow	Wallet receives multiple tokens	Multi-token withdrawal
5	Balanced Flow	Wallet sends and receives equal value	Wrapping WETH
6	Net Outflow	Wallet sends more than it receives	Lending or collateral deposit
7	Net Inflow	Wallet receives more than it sends	Collateral withdrawal

For each case, TNV is defined as the *maximum* of inflow or outflow, ignoring contract-only transfers.

5.1.3 Examples

Table 2: Examples of TNV as $\max(O, I)$

#	Outflow O	Inflow I	TNV	Notes
1	1000	0	1000	Simple send
2	0	1000	1000	Simple receive
3	500; 300	0	800	Multi-token send
4	0	800; 100	900	Multi-token receive
5	400	400	400	Balanced swap
6	600	500	600	Net outflow
7	400	500	500	Net inflow

This approach makes TNV a universal, chain-agnostic metric for measuring WalletConnect-enabled financial activity.

5.2 Fee Function Definition

The daily effective rate for each application i on day d is defined as:

$$r_i(d) = \begin{cases} \alpha\big(\mathrm{TNV}_i^{(30)}(d)\big)^{\beta-1}, & \mathrm{TNV}_i^{(30)}(d) \geq \tau, \\ 0, & \text{otherwise.} \end{cases}$$

The corresponding daily fee is calculated in USD:

$$\mathrm{Fee}_i^{\mathrm{USD}}(d) = r_i(d) \cdot \mathrm{TNV}_i(d)$$

where:

- α is a scaling factor defining overall revenue magnitude,
- β is an elasticity exponent controlling decay of the effective rate,
- τ is a billing threshold in USD,
- $\mathrm{TNV}_i^{(30)}(d)$ is the rolling 30-day cumulative TNV for app i.

At the end of each billing cycle, the accumulated USD-denominated fee can be settled in either USD-based stablecoins or WCT (conversion mechanics detailed in Section 7.1.1).

5.2.1 Threshold Parameter τ

Fees are only applied when an application's rolling 30-day Total Network Volume exceeds τ. This threshold ensures that only projects contributing meaningful economic activity are billed, while smaller or early-stage applications remain exempt until they reach sufficient scale.

$$r_i(d) = \begin{cases} \alpha\left(\mathrm{TNV}_i^{(30)}(d)\right)^{\beta-1}, & \text{if } \mathrm{TNV}_i^{(30)}(d) \geq \tau, \\ 0, & \text{if } \mathrm{TNV}_i^{(30)}(d) < \tau. \end{cases}$$

5.2.2 Example Parameterization

For illustrative purposes, the following parameter set is used:

$$\alpha = 0.0125, \quad \beta = 0.65, \quad \tau = 1{,}000{,}000.$$

For any application with

$\mathrm{TNV}_i^{(30)}(d) \geq \tau$,

is:

$$r_i(d) = 0.0125 \cdot \big(\mathrm{TNV}_i^{(30)}(d)\big)^{-0.35}$$

5.2.3 Model Output

The table below illustrates sample effective rates and corresponding monthly fees for applications at different average 30-day TNV levels, using the parameter set:

$$\alpha = 0.0125, \quad \beta = 0.65, \quad \tau = 1{,}000{,}000$$

These values are derived from the power-law fee function:

$$R(D) = 0.0125 \cdot \bar{M}(d)^{0.65}$$

- $\bar{M}(d)$ represents the application's average 30-day TNV during the billing period
- $R(D)$ represents the total fee.

Table 3: Illustrative effective rates and approximate daily fees

Application 30-Day TNV (USD)	Effective Rate (bps)	Approx. Daily Fee (USD)
\$30 B	0.027	$\approx$ \$81,000
\$10 B	0.040	$\approx$ \$40,000
\$3 B	0.060	$\approx$ \$18,000
\$1 B	0.088	$\approx$ \$8,800
\$300 M	0.135	$\approx$ \$4,050
\$100 M	0.198	$\approx$ \$1,980
\$30 M	0.302	$\approx$ \$906
\$10 M	0.444	$\approx$ \$444
\$3 M	0.676	$\approx$ \$203
\$1 M	0.993	$\approx$ \$99

These figures assume that each application's rolling 30-day TNV remains constant throughout the month, producing a steady effective rate over the billing window.

Under real network conditions, day-to-day TNV fluctuations would make total monthly fees path-dependent, but the values above serve as central estimates for a stable period.

5.2.4 Visual Analysis

Monthly Fee vs. Monthly Average TNV

A sublinear curve showing increasing fees with TNV growth, but diminishing marginal cost.

This chart illustrates the relationship between an application's average monthly Total Network Volume (TNV) and the corresponding monthly fee generated by the pricing function.

Because the function follows a power-law curve with an exponent β -1<1, the fee increases with volume but at a decreasing rate — producing a smooth, concave shape that reflects economies of scale.

The horizontal axis represents Average Monthly TNV (in billions).

The vertical axis represents the Monthly Fee (USD equivalent), computed as the aggregate of daily fees over a 30-day period.

At higher TNV values, the curve flattens, showing that large-scale applications pay substantially more in absolute terms but face progressively lower marginal costs per unit of Network Volume

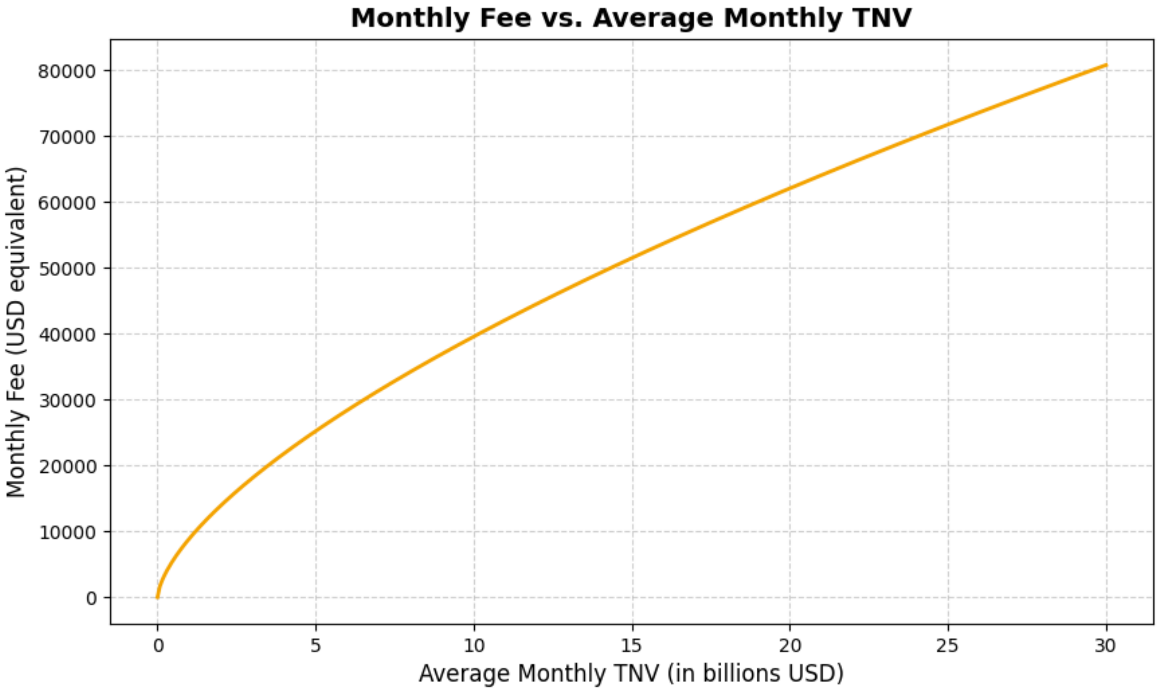

Figure 2: Monthly Fee vs. Average Monthly TNV. A sublinear curve showing increasing fees with TNV growth but diminishing marginal cost.

Effective Rate (bps) vs. Average Monthly TNV

The effective rate declines as volumes rise, favoring economies of scale and encouraging large-scale adoption.

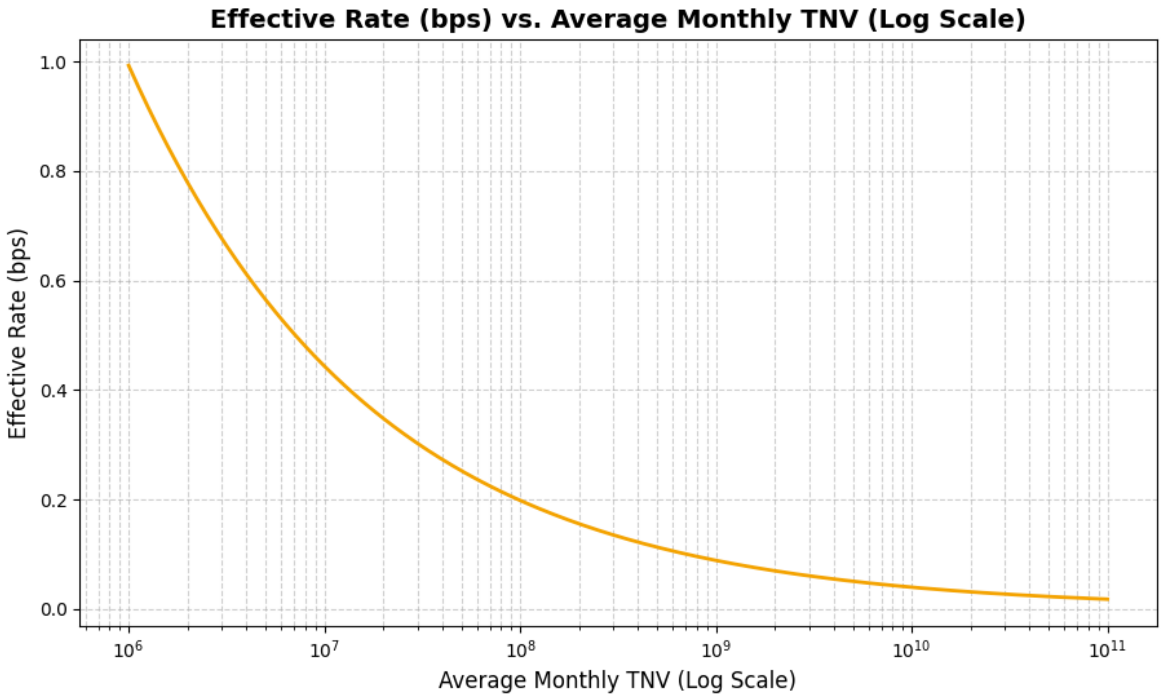

Figure 3: Effective Rate (bps) vs. Average Monthly TNV (log scale). The effective rate declines as volumes rise, favoring economies of scale and encouraging large-scale adoption.

6. Yield-Based Fee Function

While the volume-based model effectively captures value for trading and transactional activity, it is not directly aligned with the economics of lending, staking, or other yield-generating protocols.

These protocols create value through interest or yield accrual over time, rather than from discrete deposits or withdrawals.

To reflect this, WalletConnect introduces a yield-based fee function that measures realized yield facilitated through the network — fees accrue only when gains are realized (withdrawn or claimed).

6.1 Realized Yield Definition

For each withdrawal or claim event j executed by application i:

$$\text{Yield}_{i,j} = \max\big(0,\ V_{i,j}^{\text{out}} - P_{i,j}^{\text{basis}}\big)$$

$$\text{Fee}_{i,j}^{\text{USD}} = \phi \cdot \text{Yield}_{i,j}$$

Where:

$V_{i,j}^{\text{out}}$ USD-equivalent value withdrawn or claimed.
$P_{i,j}^{\text{basis}}$ Cost basis of the withdrawn portion (weighted-average or pro-rata).
ϕ WalletConnect fee multiplier (governable, bps-level).

No fee is charged if

$$V_{i,j}^{\text{out}} \le P_{i,j}^{\text{basis}}$$

6.2 Cost Basis and Partial Withdrawals

Each user position tracked through WalletConnect maintains a running weighted-average cost (WAC) — often referred to as the cost basis of the position. This cost basis represents how much capital has actually been contributed into a yield-generating position over time, accounting for multiple deposits, accrued yield, and withdrawals.

Maintaining a consistent cost basis ensures that fees are charged only on realized gains, never on principal or re-deposited funds.

Updating the Cost Basis When a New Deposit Is Made

Whenever new capital is deposited into an existing yield-generating position, the protocol updates the weighted-average cost basis using:

$$P_{\text{new}}^{\text{basis}} = \frac{P_{\text{old}}^{\text{basis}} \cdot B_{\text{old}} + V_{\text{dep}}}{B_{\text{old}} + V_{\text{dep}}}$$

Where:

$$\begin{aligned} P_{\text{old}}^{\text{basis}} &= \text{Previous weighted-average cost basis,} \\ B_{\text{old}} &= \text{Current USD value before deposit,} \\ V_{\text{dep}} &= \text{New deposit (USD)} \end{aligned}$$

This ensures that the cost basis reflects the proportional contribution of old and new capital.

—

Example (conceptual).

- A user initially deposits \$1,000 → cost basis = \$1,000.
- After 6 months at 5% APY, the position's value grows to \$1,025.
- The user deposits an additional \$500 mid-year.
- The protocol updates the weighted-average cost basis using the formula above.

Compute the old basis fraction:

$$P_{\text{old}}^{\text{basis}} = \frac{1{,}000}{1{,}025} = 0.9756, \qquad B_{\text{old}} = 1{,}025, \qquad V_{\text{dep}} = 500.$$

Now compute the new basis ratio:

$$P_{\text{new}}^{\text{basis}} = \frac{(0.9756 \cdot 1{,}025) + 500}{1{,}025 + 500} = \frac{1{,}000 + 500}{1{,}525} = 0.9836$$

Thus, 98.36% of the new position value represents contributed principal, while 1.64% represents unrealized yield accumulated so far.

By continuously tracking this ratio, WalletConnect can precisely determine the principal vs. realized-yield portion of any withdrawal, ensuring that fees apply only to actual yield—not to re-staked or re-deposited capital.

6.3 Reference-Rate Fallback

If onchain realized yield cannot be directly determined, it may be approximated using a reference rate and duration:

$$\text{Yield}_{i,j} \approx R_{\text{ref}} \cdot A_{i,j} \cdot \frac{d_{i,j}}{365},$$

where:

- $A_{i,j}$ is the USD value of the withdrawn position,
- R_{ref} is a governance-approved reference rate (e.g., onchain oracle or protocol APY),
- $d_{i,j}$ is the holding duration in days.

The corresponding fee becomes:

$$\text{Fee}_{i,j}^{\text{USD}} = \phi \cdot A_{i,j} \cdot R_{\text{ref}} \cdot \frac{d_{i,j}}{365}$$

Governance may define caps to prevent oracle manipulation or stale-position gaming.

6.4 Billing-Cycle Aggregation

For a billing period D (e.g., one month), the total fee owed by application i is the sum of all withdrawal or claim events within that window:

$$\text{Fee}_{i}^{\text{USD}}(D) = \phi \sum_{j \in W_i(D)} \max(0,\ V_{i,j}^{\text{out}} - P_{i,j}^{\text{basis}})$$

where $W_i(D)$ is the set of yield-realizing transactions for app i in period D.

6.5 Interpretation

- **Value-aligned** — Fees accrue only on realized gains, never on principal or losses.
- **Event-based** — Triggered by withdrawal or yield-claim events.
- **Governable** — Parameters R_{ref} and ϕ are adjustable via governance.
- **Transparent** — Every variable is deterministic and auditable.
- **Non-inflationary** — No fee on unrealized or negative yield.

6.6 Governance and Anti-Gaming Guards

- Dual triggers: Fees apply to both withdrawals and independent yield-claim events.
- Periodic crystallization: Governance may require periodic realization (e.g., quarterly) for long-duration positions.
- Oracle standards: Reference-rate oracles must meet freshness and deviation thresholds.
- Double-billing prevention: Certified contract tagging ensures each transaction is billed by only one model (volume or yield).

6.7 Summary Comparison

Table 4: Comparison: TNV-based vs. Yield-based fee models

Feature	TNV-Based Model	Yield-Based Model
Primary Metric	Transaction Volume (TNV)	Realized Yield / Gain
Fee Trigger	Continuous (daily activity)	Event-based (withdrawal or claim)
Economic Basis	Transaction throughput	Performance-linked returns
Best For	DEXs, bridges, payments	Lending, staking, yield vaults
Governable Params	α, β, τ	R_{ref}, ϕ
Loss Handling	Always ≥ 0 (TNV)	No fee on losses or zero gain

7. Payment Mechanics and Enforcement

To operationalize the application-side monetization framework, this section outlines the fee collection model, billing flow, grace policies, and anti-abuse measures. The system is designed to be transparent, fair, and developer-friendly — while ensuring accountability for usage of the WalletConnect infrastructure.

7.1 Monthly Billing Model

WalletConnect bills applications on a monthly basis, while accruing fees daily according to the applicable pricing function defined in Sections 5 and 6 (either the volume-based model or the yield-based model for lending/yield protocols). This design aligns value extraction with network activity in real time while maintaining predictable, once-per-month settlement.

Each application's daily fee is computed in USD, aggregated during the billing cycle, and settled at the end of the period.

7.1.1 Settlement and Currency Options

All fees are defined and calculated in USD terms, providing a stable and predictable benchmark independent of WCT market volatility. At the end of each monthly billing cycle, applications can settle their USD-denominated obligation in one of two ways:

Stablecoin settlement (USD-based).

- The application pays in a supported stablecoin (e.g., USDC, USDT, DAI).
- The payment is automatically converted to WCT at the spot price at the time of payment.
- The resulting WCT is transferred to the Revenue Vault smart contract.

Direct WCT settlement.

- The application pays directly in WCT.
- The USD-denominated fee is converted to an equivalent WCT amount at spot at the time of payment.
- That WCT is sent directly to the Revenue Vault.

In both cases, the fee remains defined in USD but is settled in WCT, ensuring predictability for developers and consistency across the network.

Rationale.

- Real-time accuracy — Fees accrue daily in line with actual network usage/value.
- Predictable cash flow — Monthly settlement cadence for accounting simplicity.
- Stable benchmark — All values expressed in USD to neutralize token volatility.
- Governable — Parameters of each pricing function are adjustable via governance.

7.2 Developer Flow & Fee Vault Mechanism

Billing and payments are managed through the WalletConnect Developer Dashboard, where developers maintain and configure their integrations via unique `project_id`s.

Payment Workflow Overview

Daily accrual.

- For each day d in the billing cycle, the protocol computes the daily fee in USD for the project, using the relevant pricing function from Sections 5 and 6.
- These daily fees accumulate automatically in the project's billing ledger.

Vault top-up and balance management.

- Each application maintains a dedicated Fee Vault, which can be pre-funded in advance to cover upcoming billing cycles.
- A Vault Management interface supports both stablecoin and WCT top-ups, automatically converting stablecoins at spot during settlement.

Developer Experience

Developers can:

- Connect a wallet directly from the dashboard,
- Review daily accruals and monthly summaries,
- Choose settlement in stablecoins or direct WCT,
- Top up balances seamlessly,
- Receive alerts as balances approach depletion.

7.3 Grace Period and Service Suspension

To maintain service continuity while enforcing accountability, WalletConnect introduces a structured grace period system tied to project balances:

- **Trigger condition** — If a project's Vault balance is insufficient at the end of a billing cycle, a 30-day grace period automatically begins.
- **During the grace period** — Developers are notified via email, dashboard alerts, and billing reminders, and may top up their Vault with stablecoins or WCT at any time.
- **If the deficit persists after 30 days** — The associated `project_id` is automatically suspended. Access to WalletConnect services — including session initiation, relaying, and network features — is paused until the shortfall is resolved.

This approach balances developer flexibility with network integrity, ensuring fair use of resources while minimizing service disruption.

7.4 Abuse Prevention and ID Rotation

To prevent abuse of the billing system through rotating AppIDs (`project_ids`), WalletConnect enforces domain binding:

- Each AppID must be registered with a verified and allowlisted set of domains (or mobile app bundle IDs).
- If a new AppID attempts to register with a domain already associated with an existing AppID that has unpaid fees or is suspended, it is automatically blocked.

This ensures continuity of billing and prevents applications from evading payments by creating "fresh" identities tied to the same web presence. While many dApps share contracts, domains represent the clearest and most enforceable point of user-facing identity. Contract-based binding is not enforced, allowing legitimate multi-UI use cases across shared protocols.

7.5 Special Cases and Exemptions

WalletConnect's billing model is designed to be inclusive yet sustainable, recognizing that not all participants contribute or extract value in the same way. The protocol supports two mechanisms to accommodate special cases.

Structural Exemptions

Certain actors are categorically exempt from fee obligations due to their network role, not economic activity, for example:

- Wallets acting as apps (e.g., custodial or embedded wallets initiating sessions via WalletConnect).

These participants are considered infrastructure providers, not value-extracting applications. They are whitelisted and provided with a subsidy to cover the fees.

Subsidized Participation via Grants

Other special cases may warrant temporary relief through subsidies or grants rather than structural exemptions:

- Non-profit entities,
- Public goods infrastructure,
- Research or education projects,
- Bootstrapping programs for early-stage or emerging-market apps.

These use cases may receive fee reimbursements or coverage from the protocol's grants budget (e.g., funded through governance-controlled treasuries), but are not excluded from billing. Such cases are evaluated individually via governance. Approved participants pay fees, but receive offsetting support as a separate funding stream.

This dual approach ensures that core infrastructure actors like wallets are supported by default, while high-impact or early-stage builders can still participate, without compromising the protocol's economic neutrality or governance-driven transparency.

7.6 Progressive Rollout

WalletConnect will introduce the fee model using a progressive rollout strategy governed by thresholds and applicability, ensuring a smooth developer experience and broad adoption. Governance can calibrate onboarding speed and scope over time.

7.7 Governance Prerequisite

Activation of the monetization framework requires community review, Council endorsement, and a formal governance vote. Without governance approval, the rollout and fee activation will not proceed.

8. Conclusion

The proposed monetization framework introduces sustainable, value-aligned fees for the WalletConnect Network while keeping end users fee-free. By combining TNV-based and yield-based models, WalletConnect can:

- Charge applications in proportion to the economic value they realize.
- Reward wallets and node operators from protocol revenue.
- Maintain governance control over key parameters and safeguards.

As network usage and financial primitives evolve, the fee functions and parameter choices can be refined via governance, but the core structure described here provides a clear and implementable starting point.

Important Note. The information provided above represents currently envisioned plans and goals intended to be the subject of a governance proposal to WCT tokenholders. The described strategies, goals, and expectations are subject to change due to rapidly evolving industry conditions and technical, legal, and regulatory developments. Reown and WalletConnect Foundation are contributors to the WalletConnect Network. Whether they or the Network are successful in achieving any particular goal may depend on factors outside of their control, including the actions of other contributors to and participants of the Network. The actual results — including the market dynamics of the WCT token — may differ materially from those anticipated. This document is not an offer or solicitation to purchase or invest in any securities, tokens, or other financial instruments.