

PRIME Token Rewards via Paragons DAO Token Staking

Sath - Yunt Capital

Objective

The relationship between Paragons DAO, sectors of the Parallel community, and speculative investors has been fractious since the events leading up to the Copper LBP launch. There are claims that the PDT token does not properly represent the value of the assets in the treasury, that PDAO is a vampire attack on Parallel, that PDAO exists only to extract value from the ecosystem, and many, many more. The primary goal of this adjustment to the tokenomics of PDT is to demonstrate that PDAO aims to have a positive symbiotic relationship with not only Parallel, but the community at large as well. Rather than being perceived as value-extractive, PDAO aims to be value-additive. Many of the larger goals for this value add are outlined in Medium articles, both published and pending, but here we focus on the relationship between the PDAO token, the assets in the treasury, and the yet-to-be-released PRIME token.

Proposal

Paragons DAO has a significant number of Prime assets, Masterpieces, and Parassets in its treasury (0x12267aefd8bb461817df348ce16c933e76c1aa0d) and as such will receive a significant percentage of airdropped PRIME tokens, as well as regularly distributed PRIME rewards. The PRIME distributed to the PDAO treasury will be a non-negligible percentage of the circulating supply of PRIME prior to the full launch of the game. As of now, there is no plan for these tokens other than to be held by the treasury and provide a liquid NAV to accompany the NAV of the illiquid NFT assets. This is good for the price of PDT, but does not serve much purpose otherwise, and does nothing to return value back to the Parallel ecosystem.

We believe that PRIME is meant to be used by players and participants in the Parallel ecosystem. To this end, PDAO being a monolithic entity that hoards a substantial percentage of the circulating supply of PRIME idle in a treasury is neither healthy nor sustainable.

We also believe that market participants should be free to choose the type of exposure they have to the Parallel ecosystem without ever giving up the ability to experience the game play and lore of the in-game universe. Perhaps an individual believes in Parallel's success, but does not want to hold thousands of dollars worth of illiquid NFTs in an uncertain market. They bond their NFTs away to PDAO in return for PDT because they want exposure to Parallel, as well as future gaming projects, while remaining liquid. We want these people to come back to the Parallel ecosystem and play the game, use the messaging system, have exposure to the success of PRIME, etc. without needing to sell their PDT or choose between holding PDT or PRIME.

In this document we propose the construction of a staking contract that takes in PDT and rewards stakers with PRIME tokens, proportional to their share of the total staked token amount, modified by a number of factors. We do this to put action behind the statement of our beliefs and to show real support for the universe that Parallel is building. This change will ensure that even individuals that have no exposure whatsoever to PDT or PDAO will benefit from the distribution of PRIME from the treasury to PDT stakers via the token sink collection/dispersal systems built into the Parallel ecosystem. These token sinks rely on healthy circulation of tokens, and benefit everyone.

Disclaimer

Before I outline this model I want to address the onslaught of "why not just use ve model" that will follow this proposal. Personally, I am hugely against arbitrarily locking tokens for years in a volatile, speculation-based market. I think it is a lazy solution to a problem that shouldn't still exist. Locking is the forfeiture of choice out of desperation created by normal market dynamics. I believe users should be rewarded for choosing to be long-term supporters of a project because of their belief in the current and future goals of a project. I also think it is perfectly reasonable for a user to have a change of heart within 4 years. People should be compensated for their active and continued belief in a project, and they should be able to leave with their gains if things change.

All that aside, I am a proponent of choice. A ve-style locking model will be available for those who prefer it.

Overview

I propose a staking model that rewards proportional shares, modified by a time-dependent function that encourages long-term staking without unstaking, and resets upon unstaking. Once again, for emphasis, the primary goal of this model is to encourage and reward continued staking, not just the amount staked.

Rather than a continuous distribution of PRIME to the staker, the PRIME rewards will be paid out in predefined intervals that are yet to be determined. Within these larger distribution intervals I propose sub-intervals (see: universeXYZ epochs) that are binary checks for staked or not staked across the full duration.

Ex: User A stakes PDT on day 3 of 10 of epoch 2. Their staking timer does not begin until the start of epoch 3. If the user then unstakes their tokens on day 7 of 10 of epoch 4, the total modifier to their APR is represented by the staking during epoch 3 only.

This is one of multiple methods designed to prevent gaming of the system where large players purchase significant amounts of PDT, stake the tokens before the checkpoint, and then sell them immediately after.

The actual distribution intervals will be defined by the distribution intervals set by the Echelon Foundation. For the purpose of this proposal, the distribution will be defined as 90 day intervals with 15 day sub-intervals.

Staking for the full duration of a sub-interval will bestow a discrete bonus to the APR modifier. This is represented mathematically as a piecewise defined step function that looks like a staircase. Example with **mock** numbers in Fig. 1.

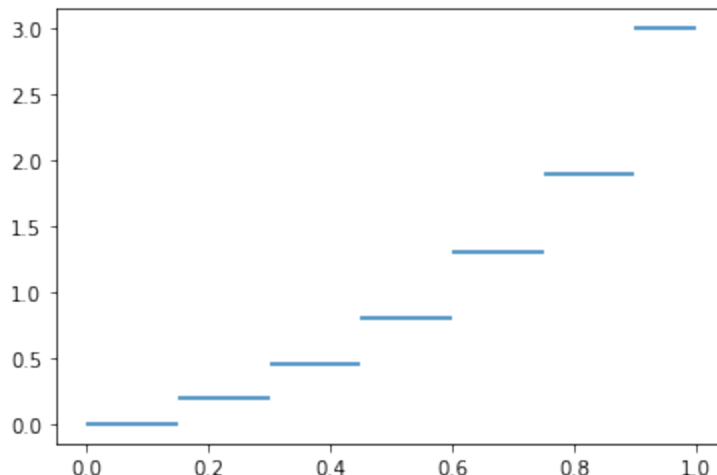


Figure 1: Mocked non-linearly increasing APR modifier for continuous epochs staked

Stakers are further rewarded if they stake for the entire 90 day interval. In addition to the full benefit of the non-linear reward weighting system for completing each sub-interval, the staker will receive a **permanent*** APR boost that carries over to the next distribution interval.

*The bonus persists so long as the tokens remain staked.

Ex: User A stakes PDT on day 1 of epoch 1 and leaves their PDT staked for the full 90 days of the distribution interval 1. At day 1 of epoch 1 for distribution interval 2 their baseline APR modifier is no longer 0, it is $(0 + x)$ from the rollover. This bonus is applied throughout all epochs and further bonuses can be compounded through completing multiple distribution intervals.

This compounding of persistent bonuses, when weighted properly, becomes a powerful motivator for continued staking. **The modifiers are far from finalized**, but will be constructed in such a way that over equal continuous durations the unlocked staking modifiers will be competitive, but not equal, with locked (ve model) option. If someone chooses to lock their PDT for 4 years their APR modifier will be net higher over the same interval than the unlocked modifier, but not by an outlandish margin.

Mechanics

To dynamically adjust reward shares of a fixed reward pool over fixed intervals we must define a scaling function. I will derive this scaling function using an example with **completely arbitrary** numbers, to illustrate the purpose of this model. For this example we assume that a 4-year locked APR is calculated on a by-epoch basis with a higher base APR than unlocked, but lower rollover. This gives noticeable advantage to 4-year lock over single distribution interval unlock, but at lower rollover it allows 4-year unlock to remain competitive.

Assumptions:

- $P = 100$ total PDT staked in staking contract.
- 4 stakers. $N = 4$.

- $T(n)$ = tokens staked by staker n .
- $M(n)$ = staking modifier for staker n .

$$T(n) = \begin{cases} T(1) = 10 \\ T(2) = 20 \\ T(3) = 25 \\ T(4) = 45 \end{cases} \quad M(n) = \begin{cases} M(1) = 3.5 \\ M(2) = 3 \\ M(3) = 1.45 \\ M(4) = 0.5 \end{cases}$$

Using the example staking APR modifiers from Fig. 1 for this example, we have staker 1 as a 4-year lock with 10% of the tokens, staker 2 as an unlocked staker who has completed their first full distribution interval with 20% of the tokens, staker 3 is unlocked and completed 5 sub-intervals with 25% of the tokens, and staker 4 is unlocked and has completed 3 sub-intervals with 45% of all staked tokens.

We can calculate an effective total scale factor S to represent the total effect of all individual APR modifiers acting on staked tokens in the contract:

$$S = \frac{P}{\sum_{n=1}^N [T(n) + M(n)T(n)]} = \frac{100}{45 + 80 + 61.25 + 67.5} = \frac{100}{253.75}$$

We can then calculate the adjusted percentage of the total rewards each staker will receive after accounting for their respective APR modifier. We'll call this R .

$$R(n) = \frac{S [T(n) + M(n)T(n)]}{P}$$

$$R(n) = \begin{cases} R(1) = 17.73 \\ R(2) = 31.53 \\ R(3) = 24.14 \\ R(4) = 26.6 \end{cases} \quad \begin{cases} 10\% \rightarrow 17.73\% \\ 20\% \rightarrow 31.53\% \\ 25\% \rightarrow 24.14\% \\ 45\% \rightarrow 26.6\% \end{cases}$$

Here we can see the change in rewards from the fixed pool of PRIME from the original model of 1 token = 1 share to a time-dependent model that allows for both locked and unlocked bonus APR rewards. As you can see, the time dependency in this example is much higher weight than the initial number of tokens staked. This rewards long-term holding, regardless of wallet size, over short-term staking to snipe PRIME distribution rewards.

Hopefully, this staking model provides a way for the average user to take a long-term interest in PDAO that provides a continuous stream of tokens with which to be an active participant in the Parallel ecosystem. Parallel supporters can bond their cards during PDT bonding events without fear that they will be losing their exposure to PRIME distributions.

Paragons DAO is here to be a positive value add to the Parallel ecosystem, not a value extractor. Hopefully this model aids in demonstrating this fact.