New Project Onboarding Procedures

Going from hello to a gauge vote.

1. Establish the pairing of assets in the pool and acquire the contracts needed.
    a. In yield bearing pairings we will need a rate provider for each asset and their token addresses.

Rate provider requirements:
Partners can reference our already reviewed repo:
https://github.com/balancer/code-review/tree/main/rate-providers

Create the issue for review request: https://github.com/balancer/code-review/issues

Examples: sDAI:
https://github.com/balancer/code-review/blob/main/rate-providers/SavingsDAIRateProvider.md
rETH:
https://github.com/balancer/code-review/blob/main/rate-providers/rETHRateProvider.md

Requirements for a rate contract are:

- The contract returns a getRate() function which is and 18 decimal fixed point number regardless of underlying tokens.
- Implements the IRateProvider interface
- If the rate provider or any underlying portion of the rate can be upgraded, changed, manipulated in any way it will not be owned by or executed by an EOA. It must be a multisig with a minimum set up of 3/5 signers. Timelock is encouraged but not required.
- Ideally is not susceptible to donation attacks and is monotonic (up only) in nature.
- Note: The rate is relative to the underlying asset, for example wsteth : stETH.

2. Creating the pool

- Use the pool creator app made by zekraken ☺ https://pool-creator.web.app/
- In the top left corner select weighted or compsableStable pool.
- Follow the usage types for swap fee, weights, and amounts. Approve tokens prior to joining the pool.
- We normally start pools with roughly 50 dollars worth of assets but this is not firm by any means.
- BE SURE TO HIT “JOIN POOL” AFTER ITS CREATED

Standard Naming
Weighted : “Balancer 80BAL/20WETH” Symbol: “80BAL/20WETH”
Stable : “Balancer wstETH/wETH” Symbol: “wstETH/wETH”

Amp factor

Can reference this framework. Typically, new LRTs 100-500, common wstETH/rETH types 1500-5000. Liquid wrappers like auraBAL 10-50 usually.

Rate Cache Duration
This is how often the rates updated in seconds. A low value is high frequency and increases gas costs; only apparent on mainnet. 90% of cases we use 10800 (6 hours) or 21600 (12 hours) – edge cases like EURO:USD we use 1 second to decrease latency.

Once created you will get a pool id and address. Use the address to create the pool's gauge using these instructions. Ethereum is a single transaction, L2s require you to make a child gauge on the respective chain, then a root gauge on mainnet to correspond to the child address. This is how rewards stream from L1 to L2.

Partner can post on the forum, reference their pool address, gauge address(es) and the following specification:

Technical Specification:
Transaction: The Balancer Maxi LM
Multisig eth:0xc38c5f97B34E175FFd35407fc91a937300E33860 will interact with the GaugeAdderv4 at 0x5DbAd78818D4c8958EfF2d5b95b28385A22113Cd and call the addGauge function with the following arguments:

gauge(address): "ROOT GAUGE ADDRESS FROM MAINNET"

gaugeType(string): "NETWORK THE POOL IS ON"

Create fresh wallet and use the gas station to get eth for gas - ask in the maxi chat
.01 eth in each asset
Send URL to zekraken to fix the FE and WL (token logo, address). PNG square file of your logo
Pass API to zen.