Distribution of staking incentives (COMP/StkAAVE)

- When a new expiry is created, a `PendleYieldTokenHolder` contract is also deployed alongside the OT and XYT contract
- The `PendleYieldTokenHolder` will hold all the aToken/cToken
- All the transfers of aToken/cToken is done from/to the `PendleYieldTokenHolder`
- There will be COMP/StkAAVE rewards accrued in the `PendleYieldTokenHolder`
- Whenever there is new COMP/StkAAVE rewards coming into the `PendleYieldTokenHolder`, it will be distributed equally to the current OT holders
  - At time `t0`: total amount of OT is `totalOT_0`, a new user `A` mints `balanceA` OTs
  - At time `t1`: There is `rewards_0` coming in, A should get:
    - `balanceA/totalOT_0 * rewards_0`
  - At time `t2`: There is `rewards_1` coming in, A should get:
    - `balanceA/totalOT_1 * rewards_1`
    - In total: A should get `balanceA * (rewards_0/totalOT_0 + rewards_1/totalOT_1)`
  - As such, we can generalise it and let:
    - `L(t+1) = (rewards_0/totalOT_0 + rewards_1/totalOT_1 + ... + rewards_t/totalOT_t)`
    - Then, rewards for a user A with `balanceA` for holding OT from `t1` to `t2` is:
    - `balanceA * ( L(t2) - L(t1) )`
- As such, before any action that changes the totalOT, we will then need to:
  - claim the COMP/StkAAVE rewards

    - update L(t), based on the previous totalOT, and the amount of rewards that came in since the last L(t) update
- Another note is that this way of accounting the rewards is exactly the same as how we account for the interests for a generic pool, especially when the "rewards/interests" do not grow by itself (same as a Compound Market, or Compound liquidity mining contract)
    - Therefore, we try to keep the implementation to be as similar as possible, to reduce potential bugs and mistakes