

# M1 Trading Automation

M1 Finance automates the construction and maintenance of custom investment portfolios. After the user creates their Pie-Based portfolio, M1 utilizes its proprietary trading system to automate all trades. The user never inputs or executes any trades manually. As money is deposited and withdrawn from the account, M1 dynamically rebalances a users portfolio by pushing it towards the desired allocations. M1 aggregates all trades and executes them during a single trading window. All trading activity for the day will be completed during this window. M1 utilizes event driven automation. There are distinct events the automatically trigger trading on a users account: initial investment, rebalances, deposits, withdrawals, and liquidations.

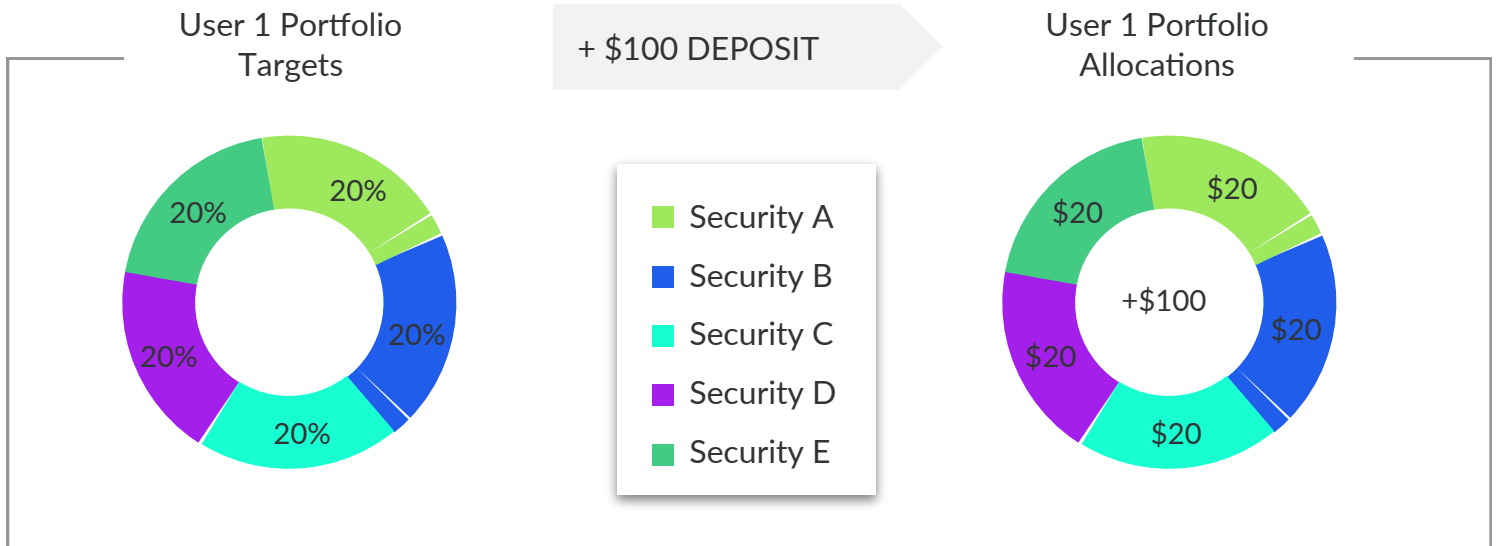
## Contents

Initial Account Funding	1
Rebalance	3
Cash Inflows & Dynamic Rebalance	6
Cash Withdrawals & Dynamic Rebalance	9
Changing Targets, Adding Slices and Liquidations	
Changing Targets	12
Adding Slices	15
Liquidations	22

# Initial Account Funding

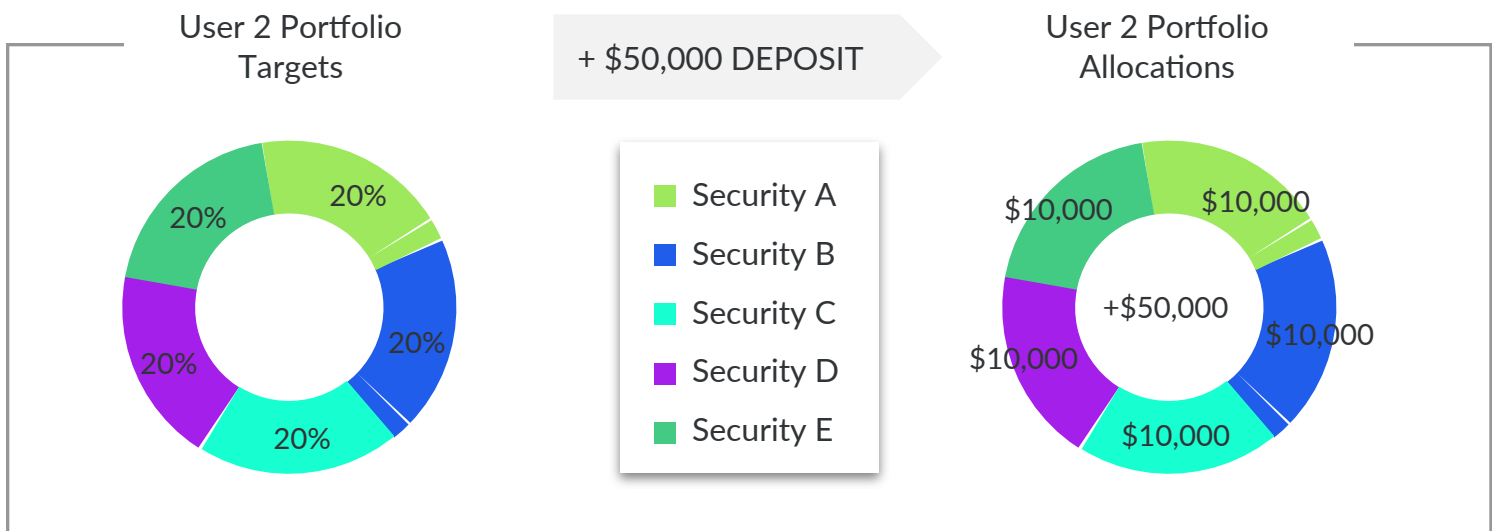
M1 allows users to create a portfolio of exchange listed securities on a percentage basis. The initial cash flow will be allocated to each security in alignment with the user specified allocations using fractional shares to invest down to the penny.

**Example 1:** User 1 defined their portfolio with five equally weighted securities and invests \$100.



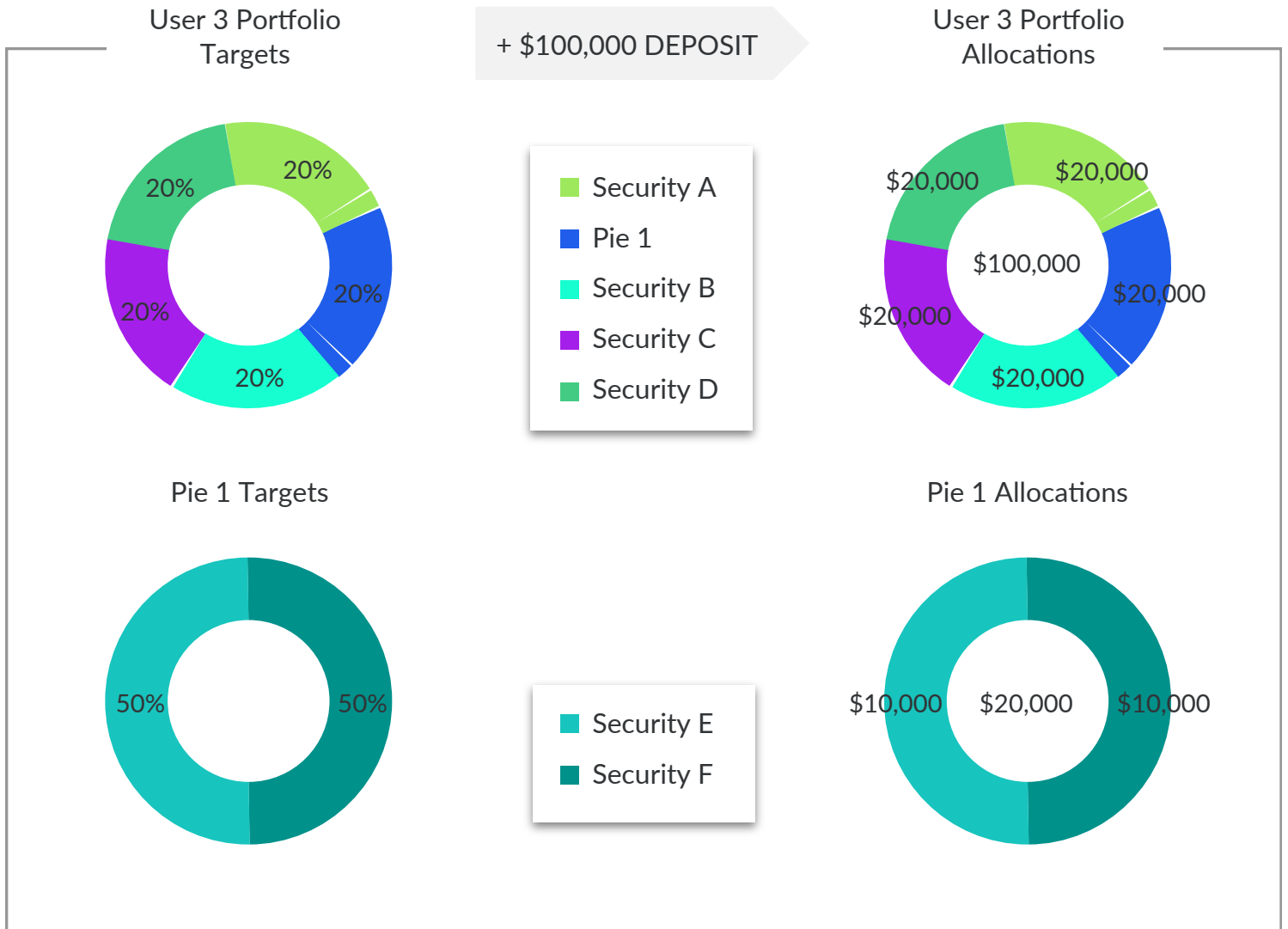
**Result:** User buys \$20 of each slice.

**Example 2:** User 2 defined their portfolio with five equally weighted securities and invests \$50,000.



**Result:** User buys \$10,000 of each slice.

**Example 3:** User 3 has created a portfolio consisting of five equally weighted slices and one of the slices represents a Pie containing two equally weighted securities. They invest \$100,000.



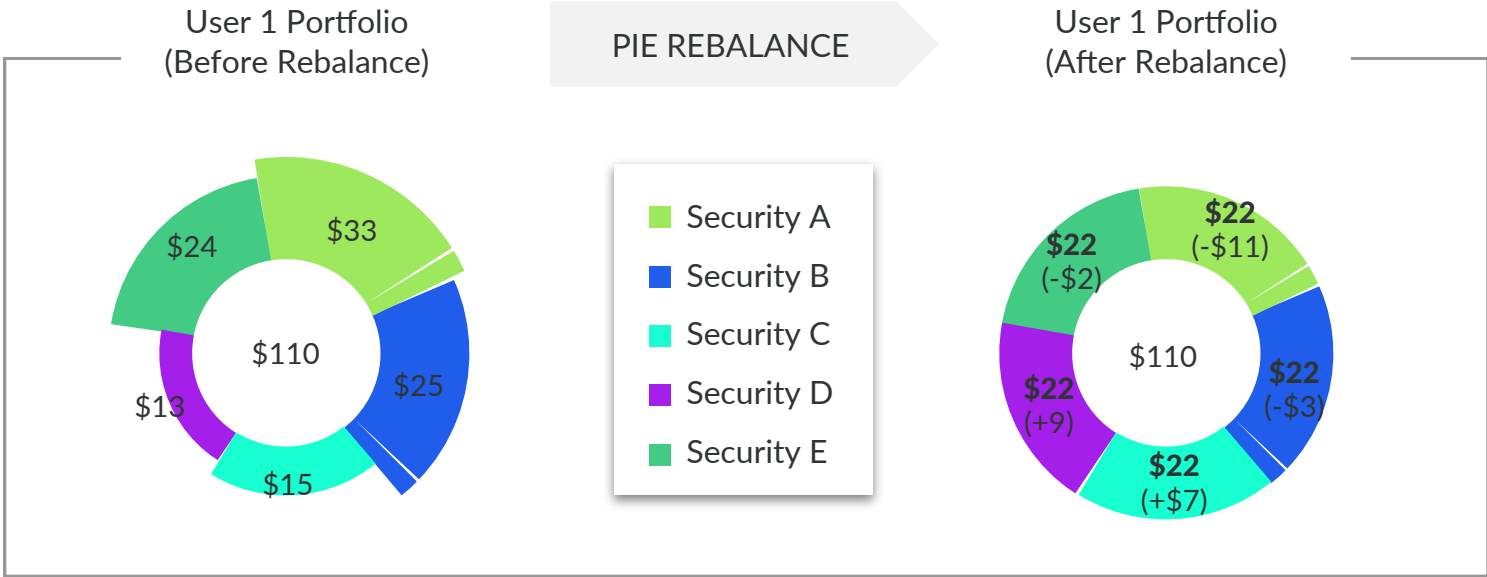
**Result:** User buys \$20,000 of each slice. The slice referencing Pie 1 has the \$20,000 purchase split evenly between Security E and Security F.

# Rebalance

M1 allows users to rebalance their Pies to bring them back to their original target allocations. Rebalances can be triggered on any Pie (including the total portfolio).

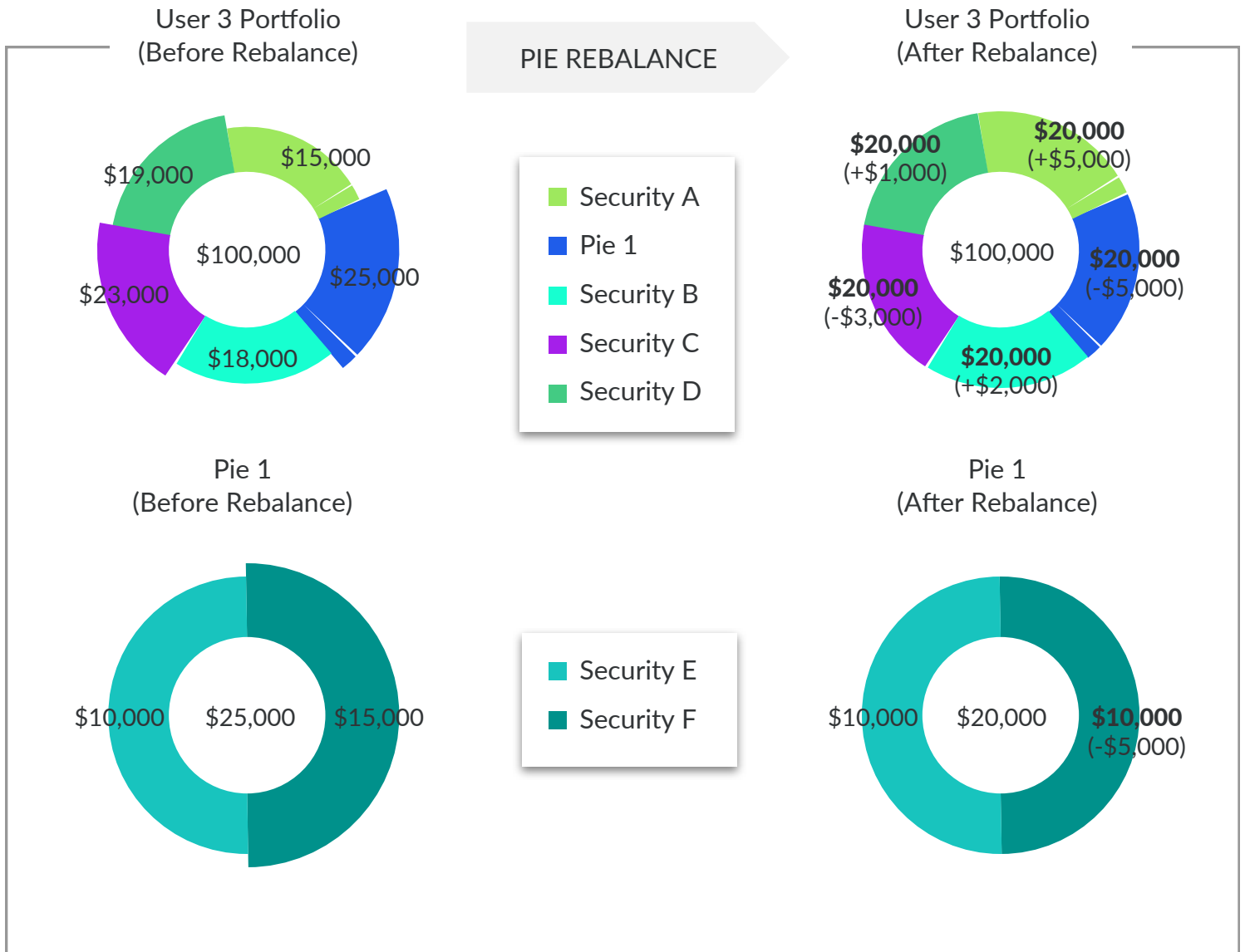
**Why does this happen?** As markets go up and down, your portfolio will drift from target allocations. Rebalancing periodically can keep investors aligned with their target allocations and investment goals. This will reduce risk exposure relative to your target portfolio. A portfolio that goes a long time without any rebalance can have risk exposures that differs from a user's initial plan.

**Example 1:** User 1 has a portfolio with five equally weighted securities which they rebalance.



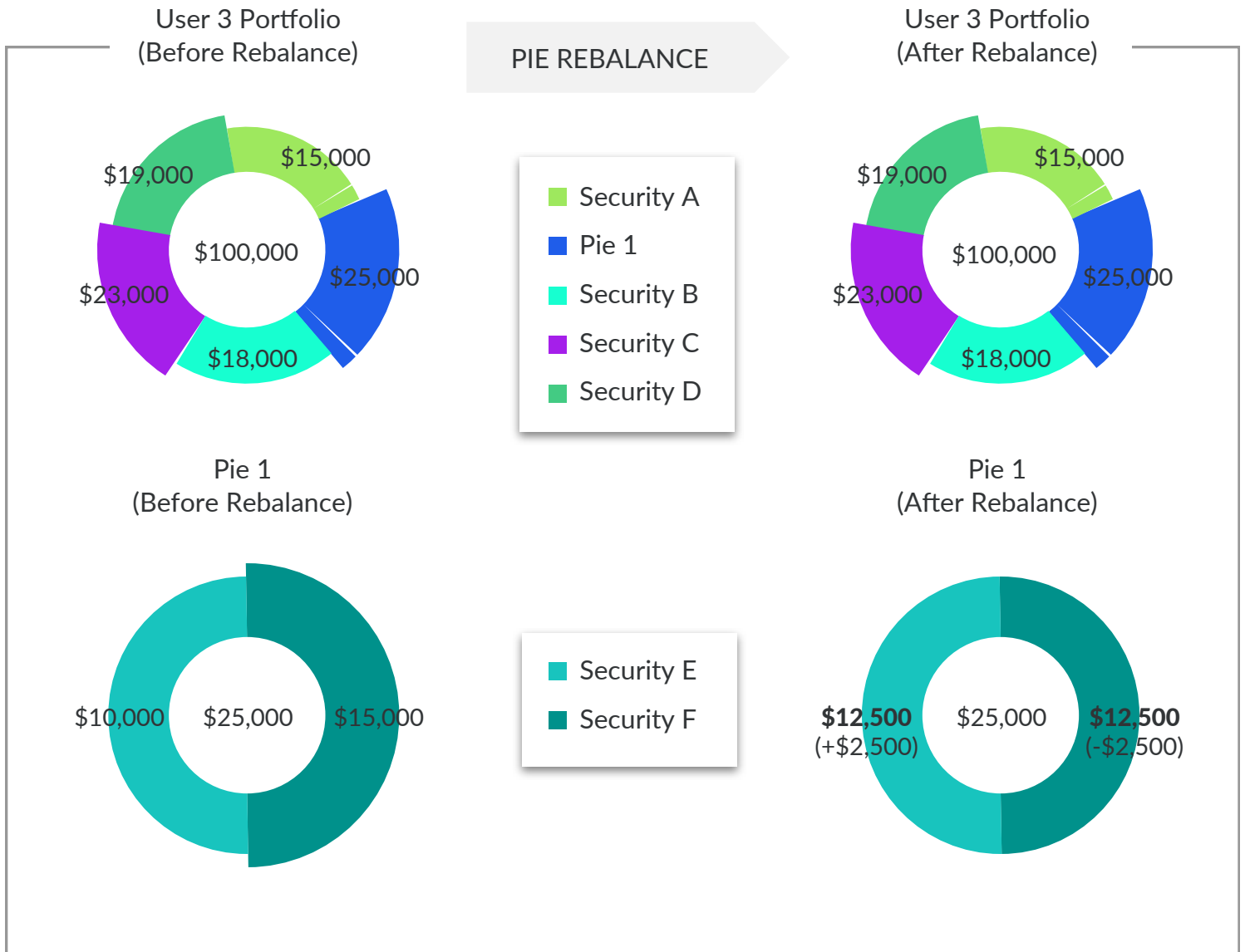
**Result:** The rebalance gets all slices to a \$22 market value. To reach this rebalance, each slice that is worth more than \$22 has a sale of the underlying security. Each slice worth less than \$22 has a purchase of the underlying security.

**Example 2:** User 3 rebalances his entire portfolio which consists of five equally weighted slices and one of the slices is comprised of a Pie with two equally weighted securities. He rebalances his portfolio.



**Result:** The rebalance gets all slices to a \$20,000 market value. To reach this rebalance each slice worth more than \$20,000 has a sale of the underlying security while each slice worth less than \$20,000 has a purchase of the underlying security. Pie 1 is rebalanced recursively after its target value was determined by the total portfolio rebalance target values.

**Example 3:** User 3's portfolio consists of five equally weighted slices and one of the slices is comprised of a Pie with two equally weighted securities. He rebalances only the nested Pie (Pie 1).

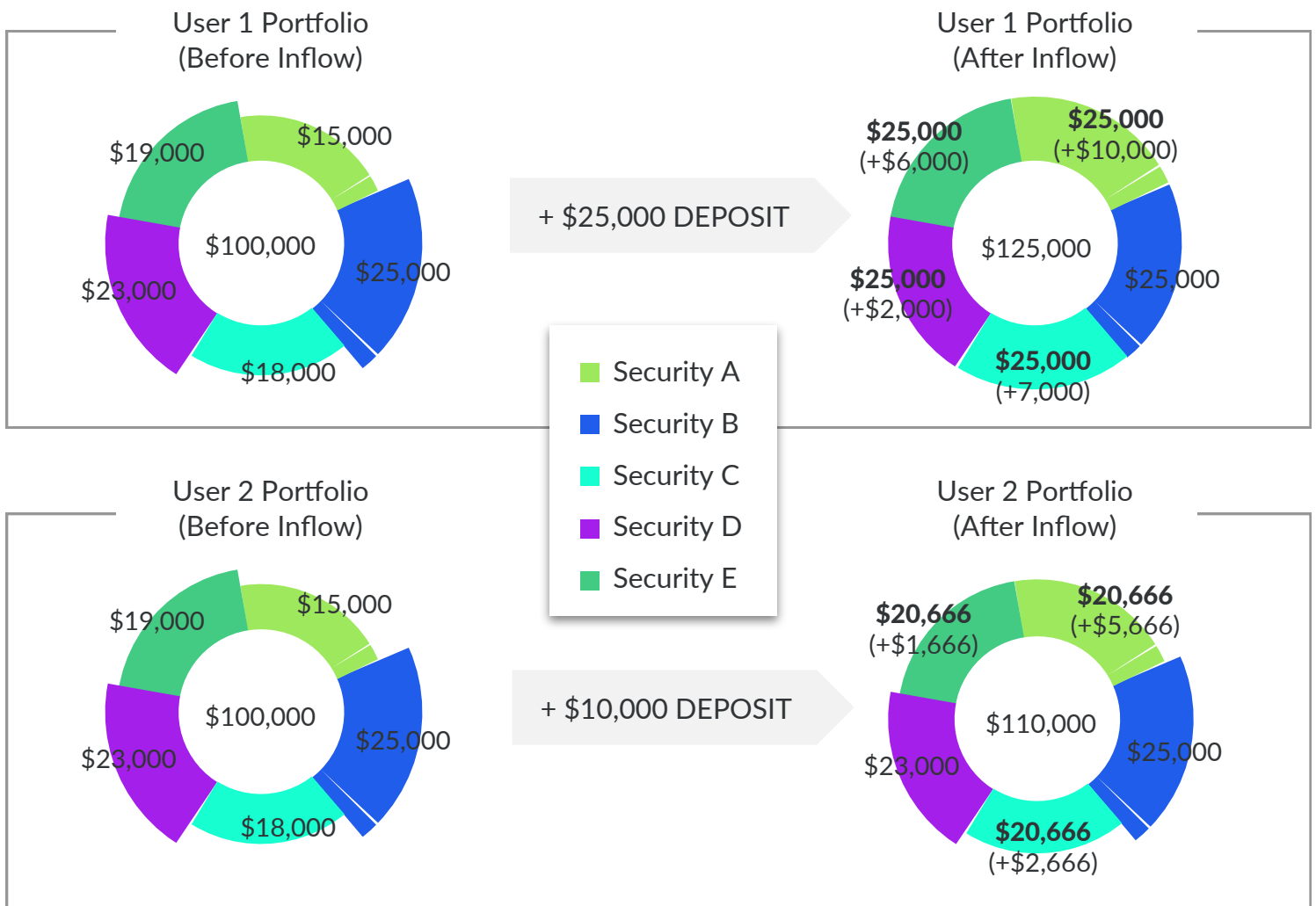


**Result:** The rebalance is only for Pie 1. So all slice values at the total portfolio level remain unchanged and the securities underlying Pie 1 are set to their target values of an even 50/50 split. Security E has a \$2,500 purchase while Security F had a \$2,500 sale.

## Cash Inflow & Dynamic Rebalance

When user accounts receive cash inflows from a dividend payment, a recurring payment is scheduled, or a user manually depositing funds into their non-dynamic M1 account, M1 will actively push accounts towards their portfolio targets. M1 will buy the most proportionately underweight slices first and then mechanically buy securities until the cash inflow is completely deployed or the portfolio is rebalanced. M1 will never sell securities on a cash inflow.

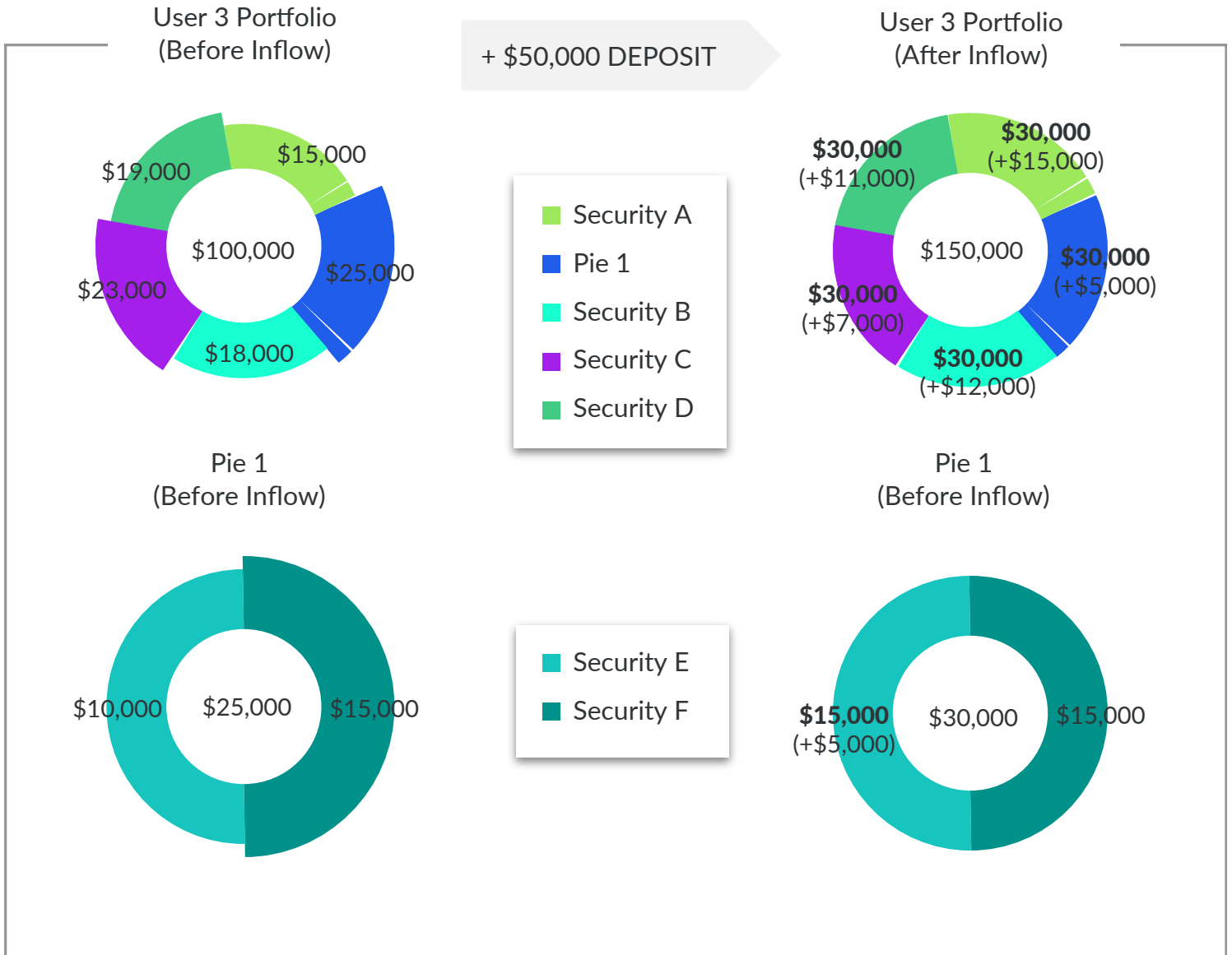
**Example 1:** User 1 and User 2 both have portfolios with 5 equally weighted securities. User 1's account received a cash inflow large enough to push his account to its target weights and User 2's account receives a cash inflow not large enough to push his account to its target weights.



**Result:** User 1's cash inflow of \$25,000 is enough to get all slices to their target allocations. Each slice purchases enough of its underlying security to get a target value of \$25,000. (Since Security B already had a market value of \$25,000 there will be no purchase or sell.)

User 2's cash inflow of \$10,000 is not enough to get all slices to their target allocations so the purchase starts with the most underweight slice (Security A) and utilizes the entire deposit to purchase securities to get the User's portfolio values closest to the target allocations.

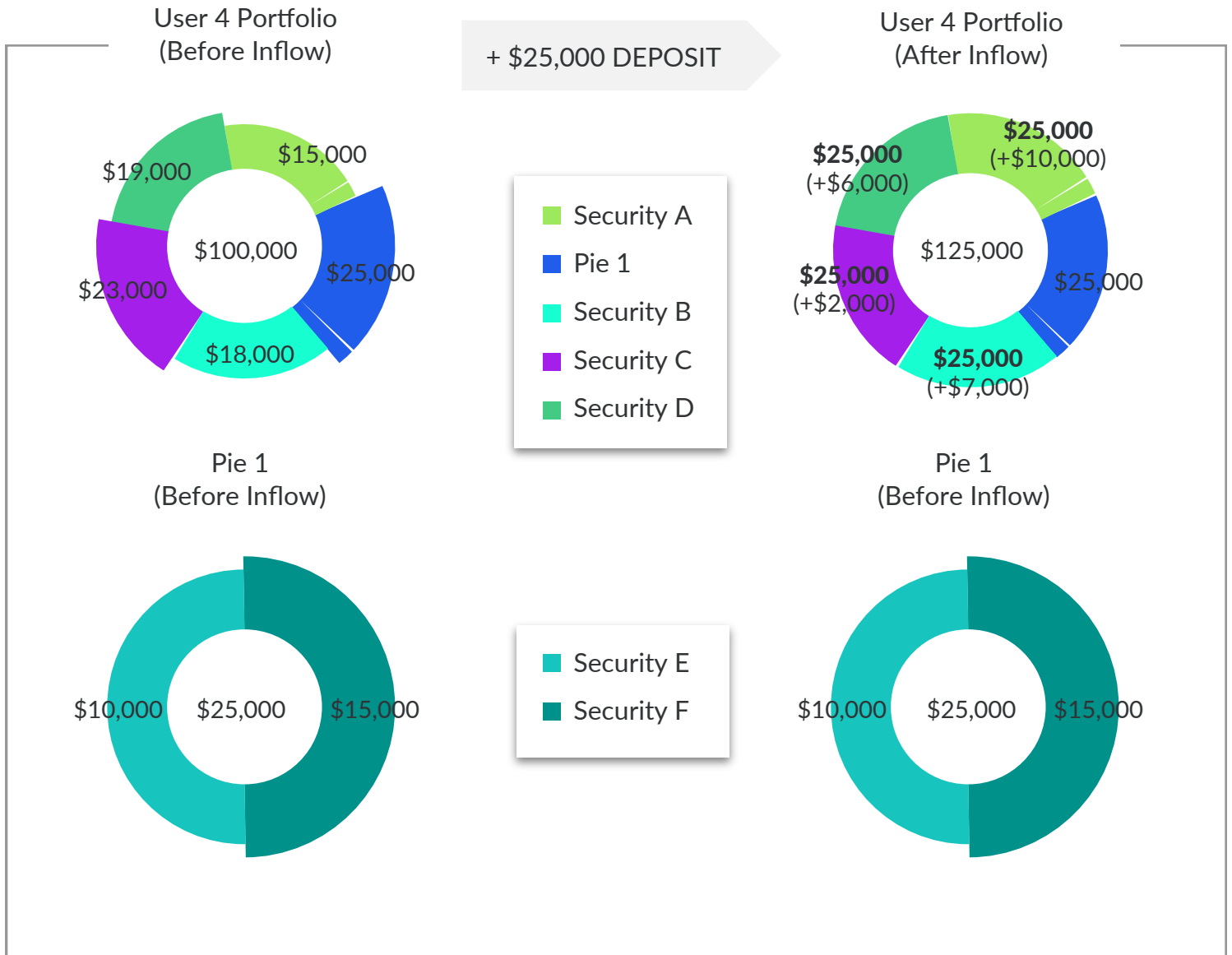
**Example 2:** User 3's portfolio consists of five equally weighted slices and one of the slices is comprised of a Pie with two equally weighted securities, and a cash inflow that flows into the nested Pie (Pie 1).



**Result:** User 3's cash inflow of \$50,000 is enough to get all slices to their target allocations. Each slice purchases enough of its underlying security to get a target value of \$30,000. The slice referring to Pie 1 receives a cash flow of \$5,000 from the Top Level Portfolio deposit calculation, this \$5,000 is used for a purchase of Security F to get Pie 1 to its target allocation.



**Example 3:** User 4's portfolio consists of five equally weighted slices and one of the slices is comprised of a Pie with two equally weighted securities. They have a cash inflow that does not flow into the nested Pie (Pie 1).

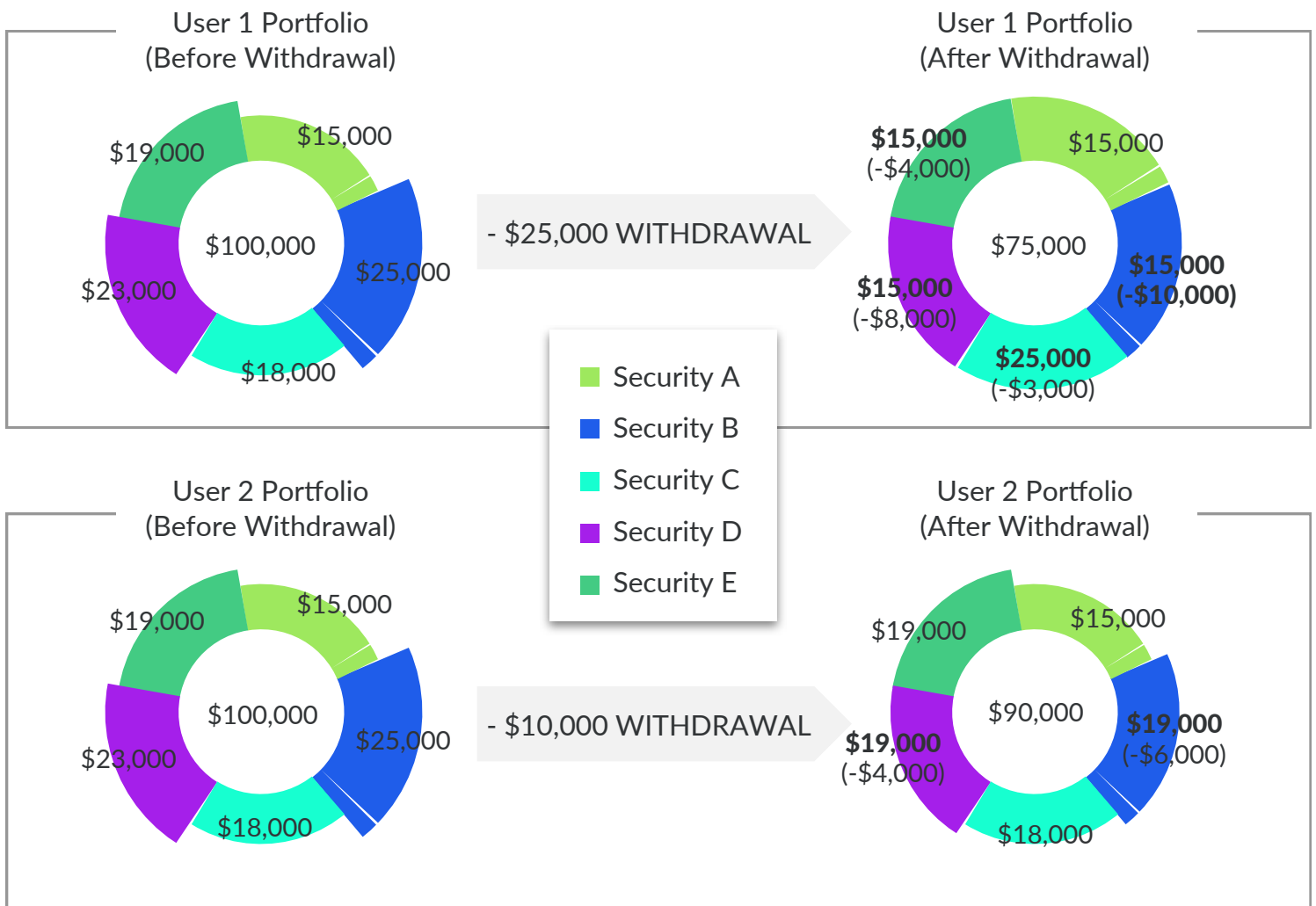


**Result:** User 4's cash inflow of \$25,000 is enough to get all slices to their target allocations. Each slice purchases enough of its underlying security to get a target value of \$25,000. The slice referring to Pie 1 receives no cash flow from the Top Level Portfolio deposit calculation, so Pie 1 will not have any purchases.

## Cash Withdrawal & Dynamic Rebalance

When M1 accounts receive cash withdrawal requests from a user's scheduled periodical withdrawals, or a user manually requesting funds from their non-dynamic M1 account M1 will actively push accounts towards their portfolio targets. M1 will sell the most proportionately overweight slices first and then mechanically sell securities until the cash outflow is completely covered or the portfolio is rebalanced. M1 will never buy securities on a cash outflow.

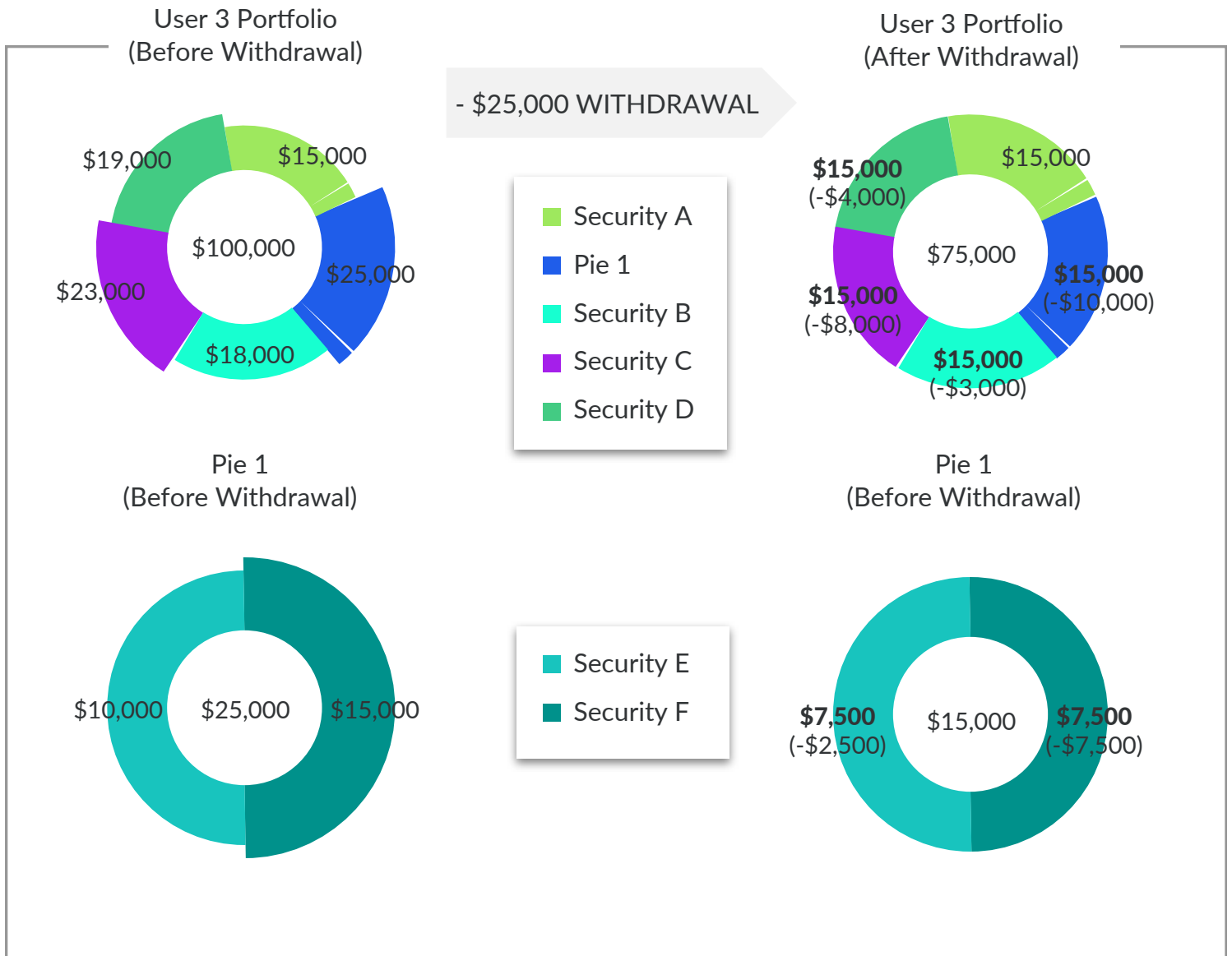
**Example 1:** User 1 and User 2 both have portfolios with 5 equally weighted securities. User 1's account received a cash withdrawal request large enough to push his account to its target weights and User 2's account receives a cash withdrawal request not large enough to push his account to its target weights.



**Result:** User 1's cash withdrawal of \$25,000 is enough to get all slices to their target allocations. Each slice sells enough of its underlying security to get a target value of \$15,000.

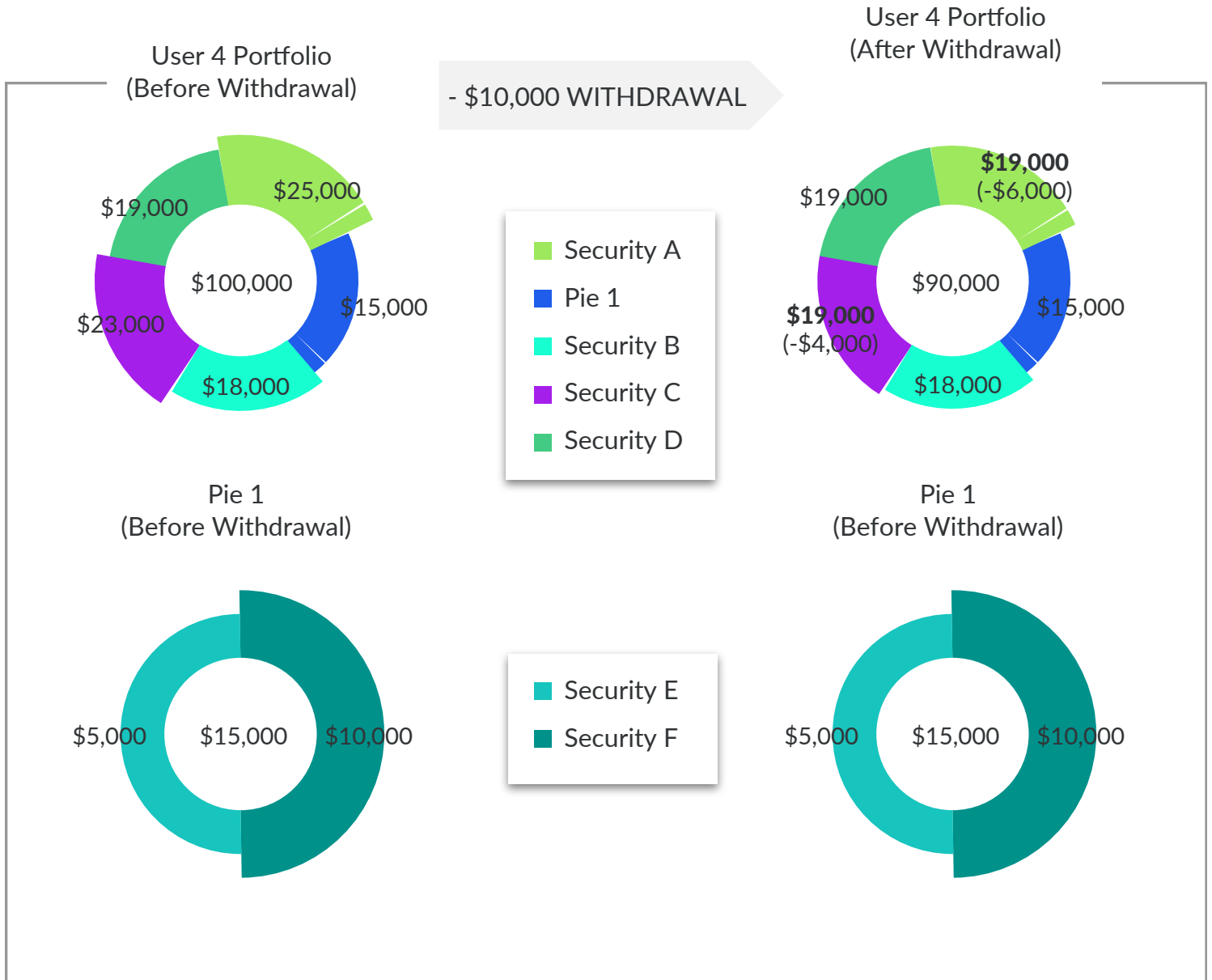
User's 2's cash withdrawal of \$10,000 is not enough to get all slices to their target allocations so the sales start with the most overweight slice and mechanically utilizes the entire withdrawal to sell securities to get the User's Portfolio values closest to the target allocations.

**Example 2:** User 3's portfolio consists of five equally weighted slices and one of the slices is comprised of a Pie with two equally weighted securities, he has a cash withdrawal request that effects his nested Pie.



**Result:** User 3's cash withdrawal of \$25,000 is enough to get all slices to their target allocations. Each slice sells enough of its underlying security to get a target value of \$15,000. The slice referencing Pie 1 receives a cash withdrawal of \$10,000 from the top level portfolio withdrawal calculation, the \$10,000 withdrawal is split between a \$7,500 sale of Security E and a \$2,500 sale of Security F.

**Example 3:** User 4's portfolio consists of five equally weighted slices and one of the slices is comprised of a Pie with two equally weighted securities. He has a cash withdrawal request that does not affect his nested Pie.



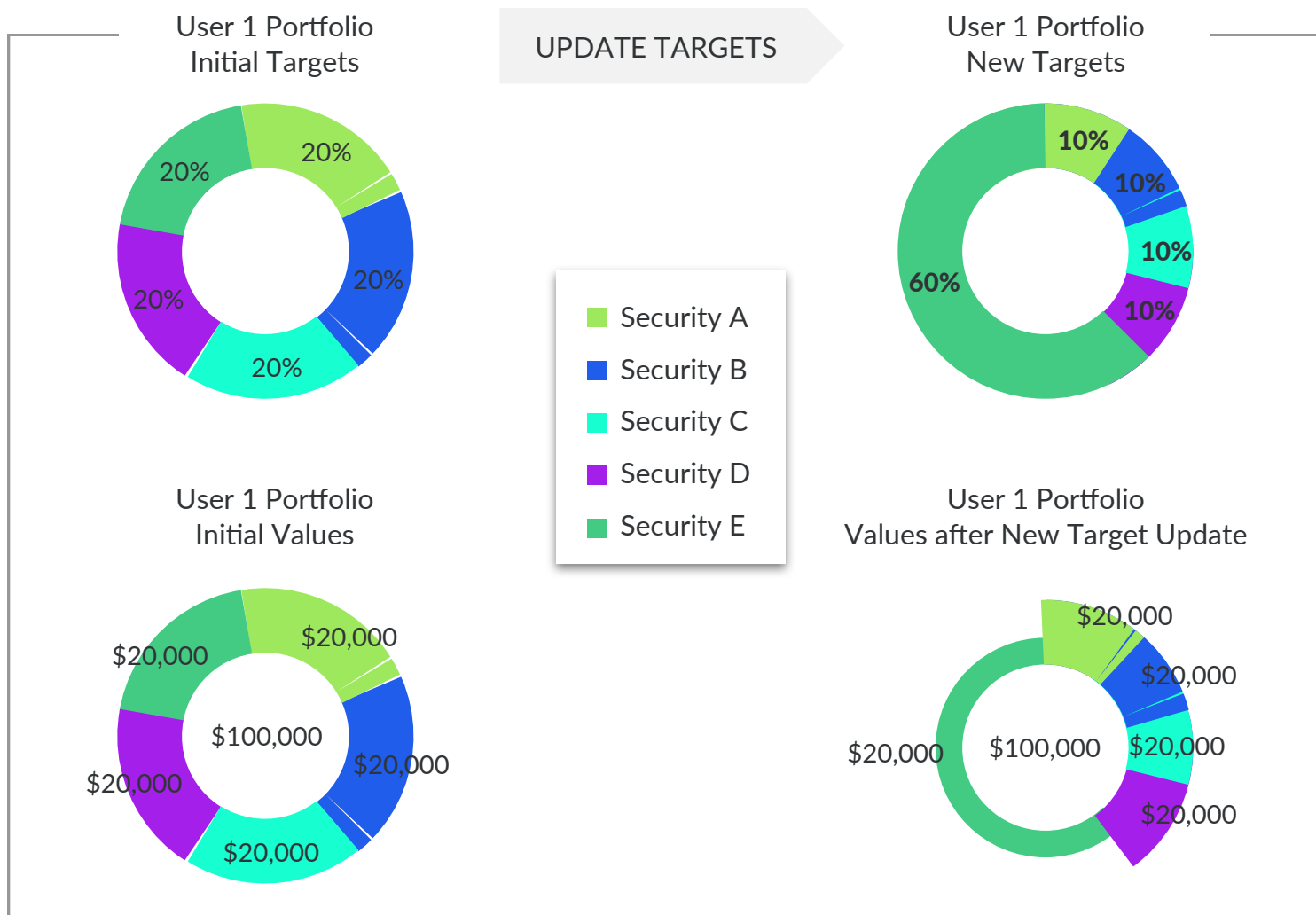
**Result:** User 4's cash withdrawal of \$10,000 is not enough to get all slices to their target allocations so the sales start with the most overweight slice and accrues the entire withdrawal to sell securities to get the user's portfolio values closest to the target allocations. The slice referencing Pie 1 receives none of the cash withdrawal of \$10,000 from the top level portfolio withdrawal calculation, Pie 1 will have no trading activity.

## Changing Targets, Adding Slices, and Liquidations

Solely changing current targets or adding slices does not cause any trades to occur. The changes made will start to be pushed towards once a user either rebalances his/her account or has a cash event. The only time changing a target allocation triggers a trade is if a slice that currently has a dollar value greater than zero is set to a 0% target allocation, this causes a liquidation. A liquidation will set off an immediate sale of the security or Pie held in the slice and the proceeds will be a cash inflow into the liquidated slice's Pie.

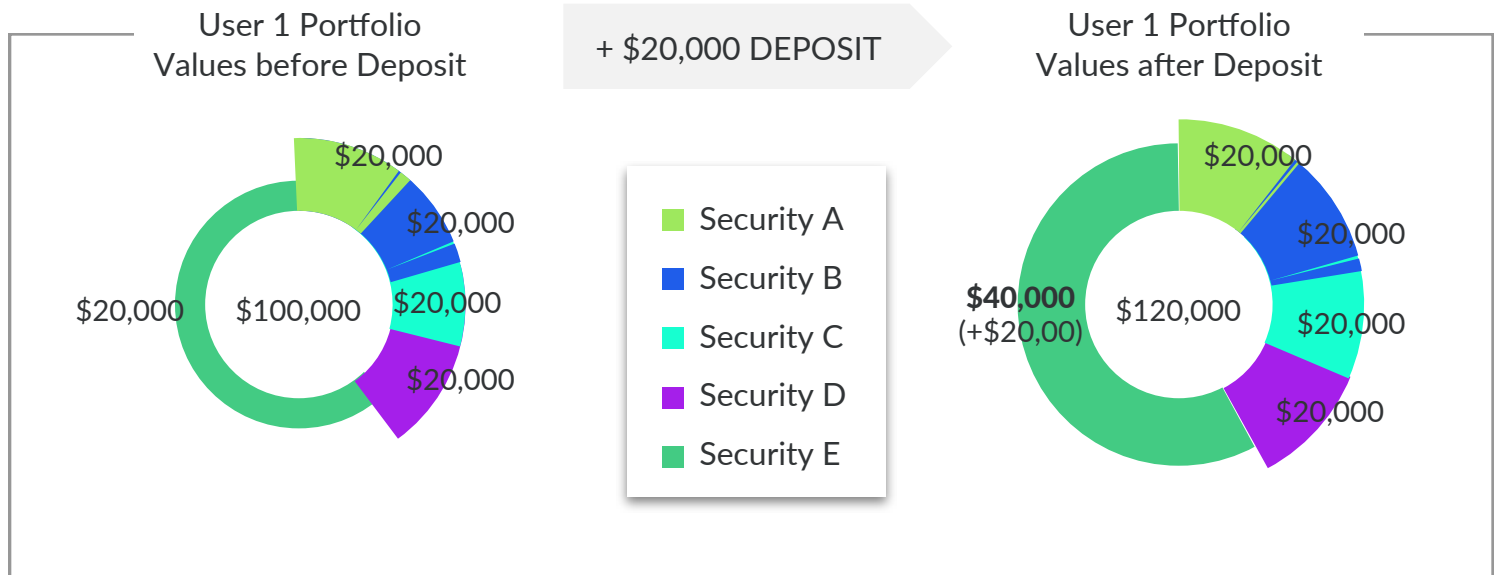
**Example 1 (Changing Targets):** User 1 has a portfolio with 5 equally weighted securities. (1) User 1 decreases four of the slices to 10% and increases the last slice to 60%. (2) User 1 makes a cash deposit, (3) a cash withdrawal and (4) then later rebalance their portfolio.

### 1. User 1 makes target allocation change



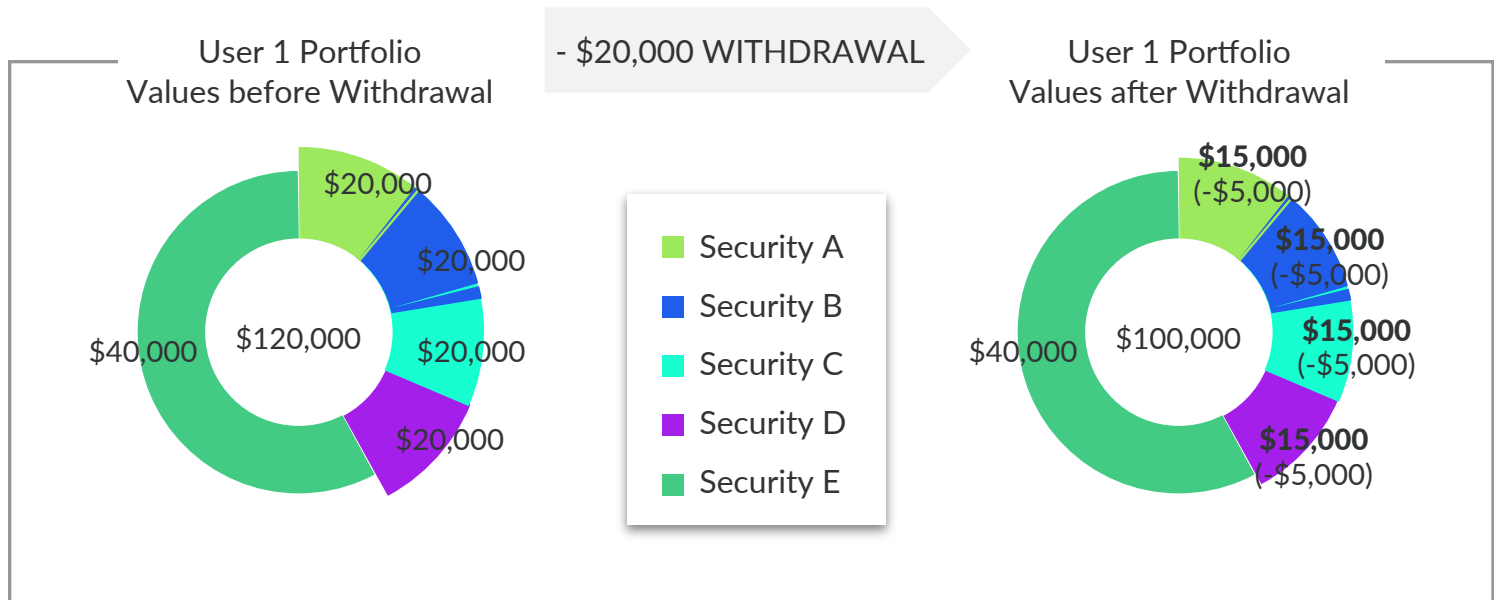
**Result:** The update of target allocations cause no immediate trading activity.

2. User 1 makes a cash deposit



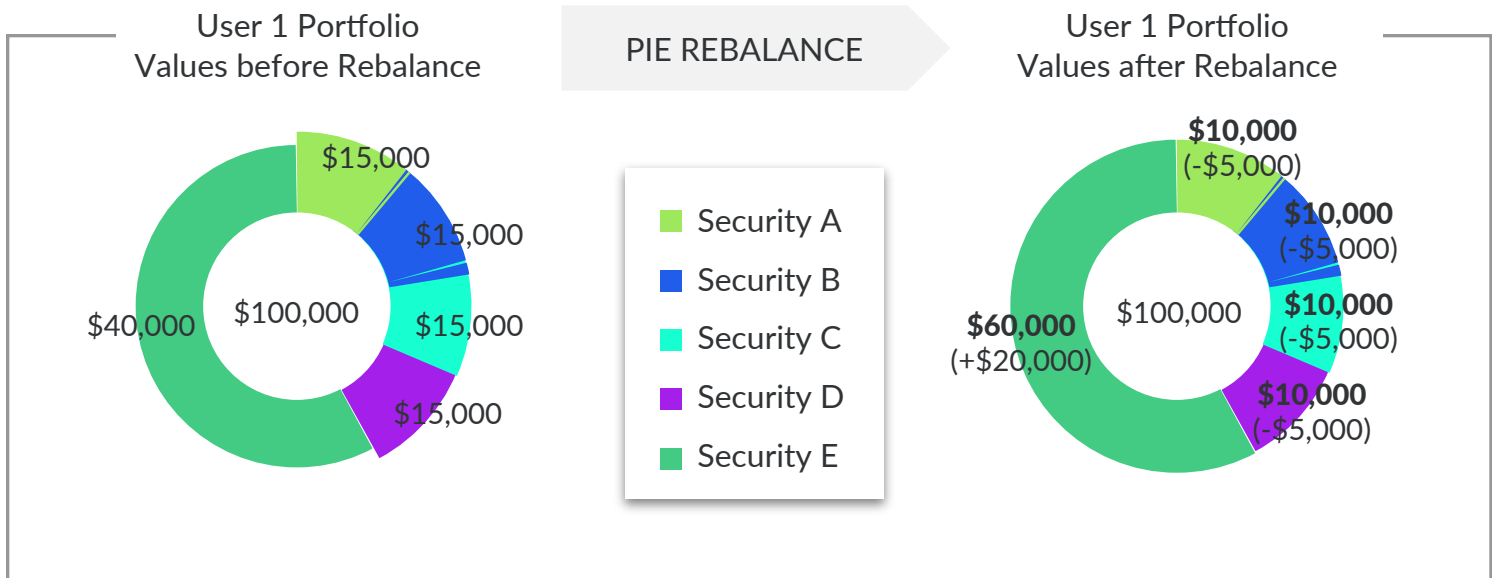
**Result:** The \$20,000 cash inflow will be used to push User 1's portfolio towards the new portfolio targets with a \$20,00 purchase of Security E.

3. User 1 makes a cash withdrawal



**Result:** The \$20,000 cash withdrawals will be used to push User 1's portfolio towards the new portfolio targets with with \$5,000 sales of Security A,B, C, and D.

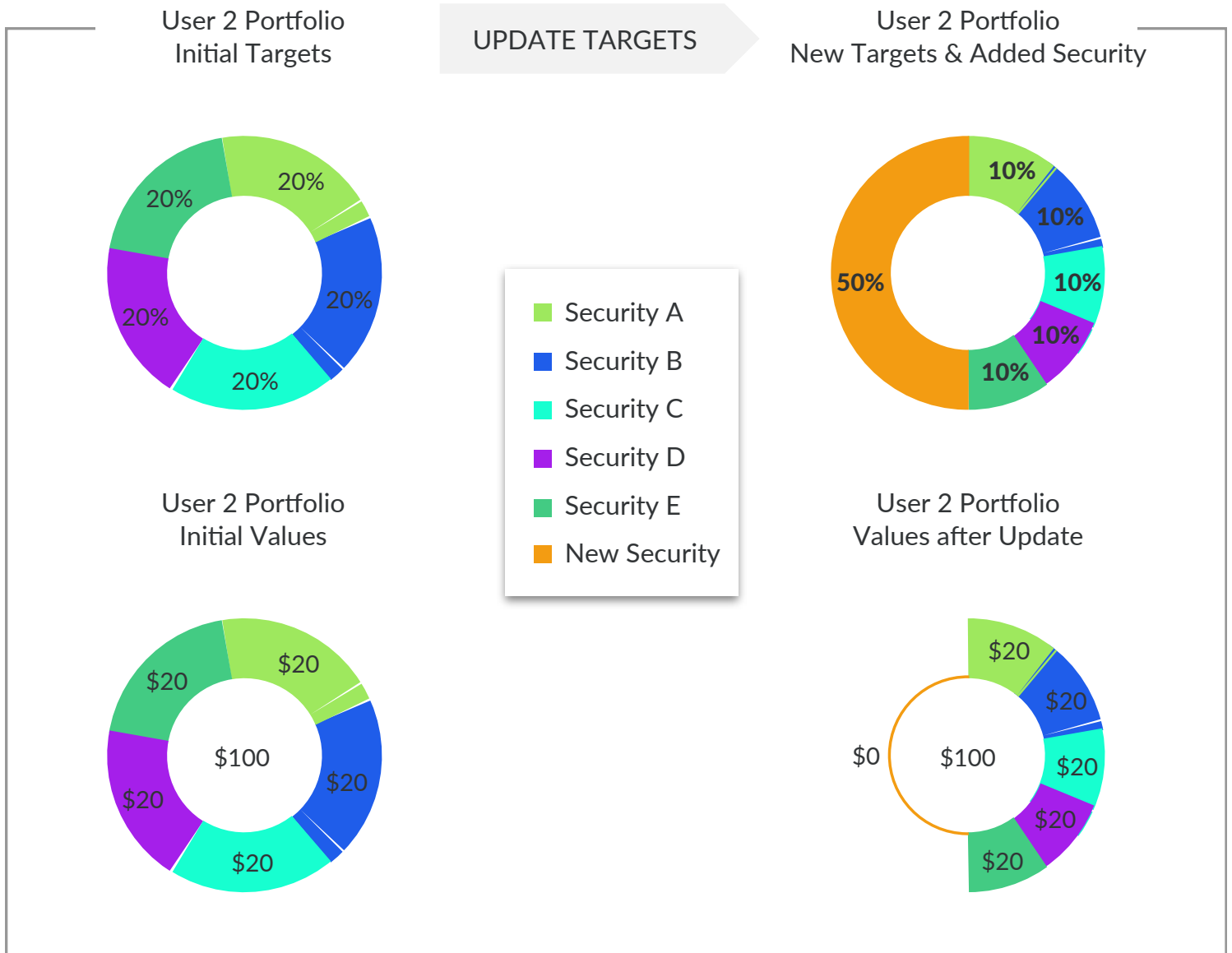
4. User 1 makes a rebalance to portfolio



**Result:** The rebalance pushes User 1's portfolio values to the new target allocations.

**Example 2 (Adding Slices):** User 2 has a portfolio with 5 equally weighted securities. (1) User 2 adds a 6th security to his portfolio at a 50% target allocation. (2) User 2 makes a cash deposit, (3) a cash withdrawal and (4) then later rebalance their portfolio.

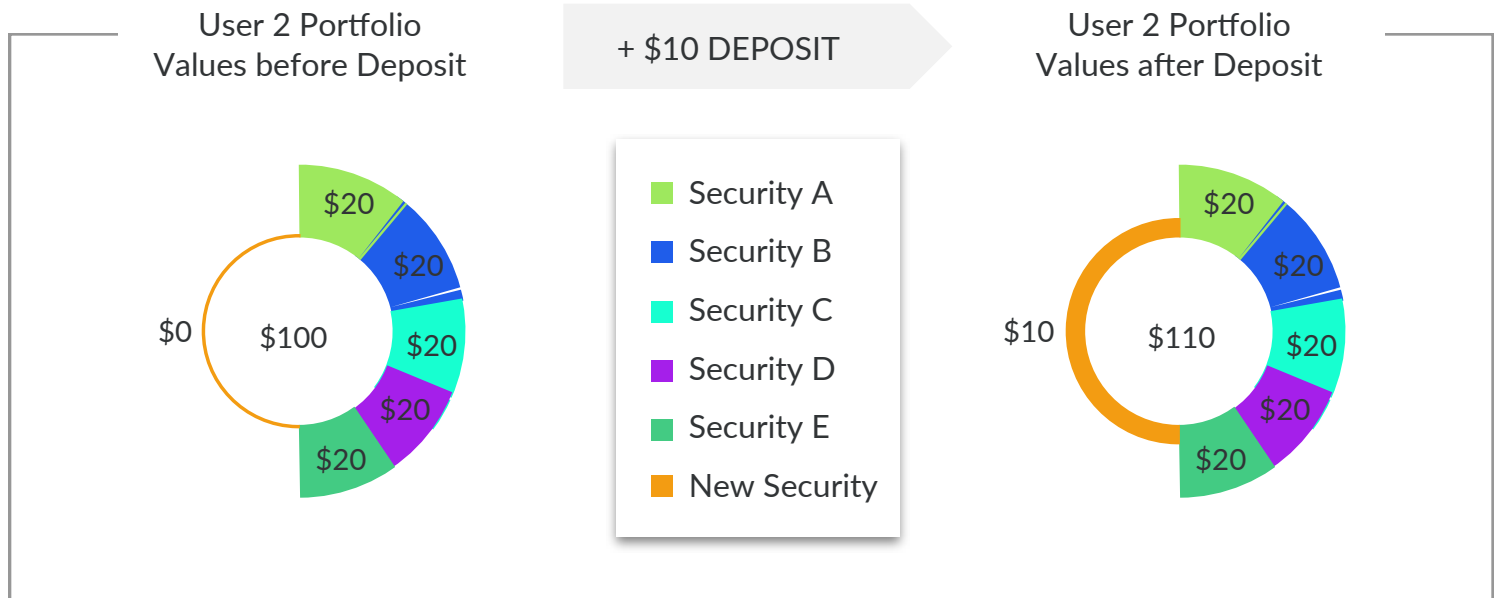
1. User 2 makes target allocation change



**Result:** The update of target allocation cause no immediate trading activity even with the new security added.

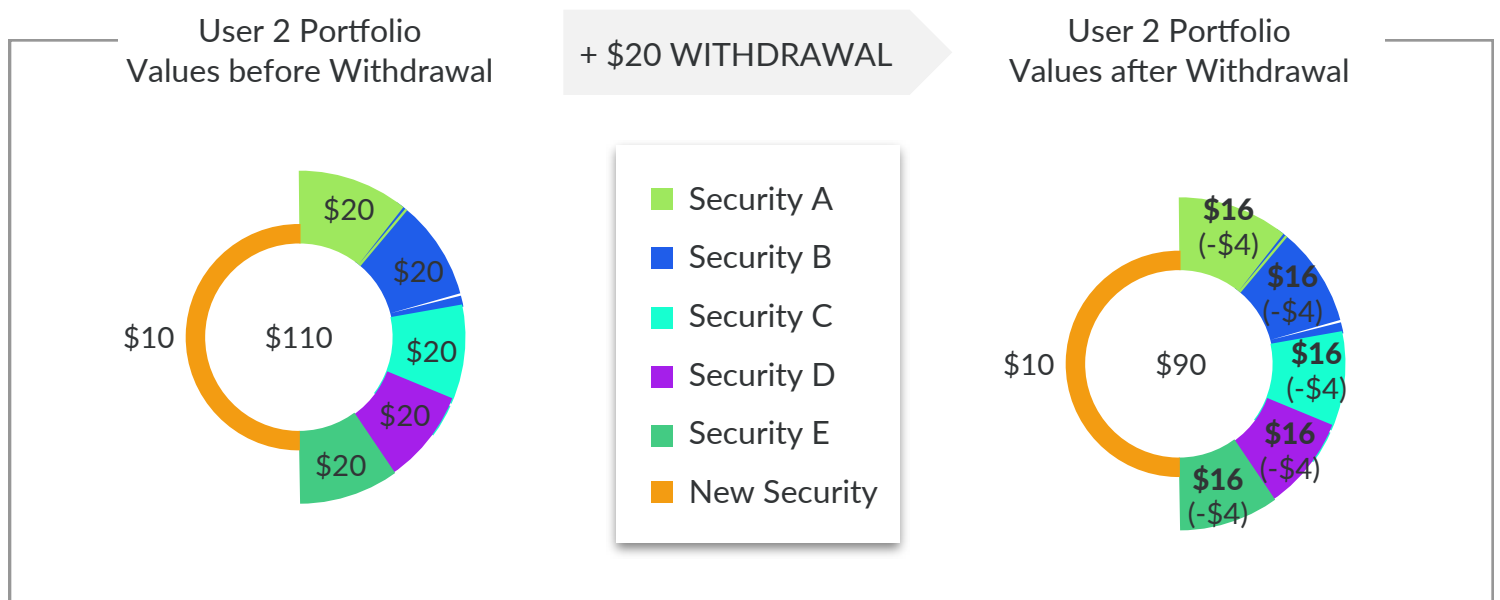


2. User 2 makes cash deposit



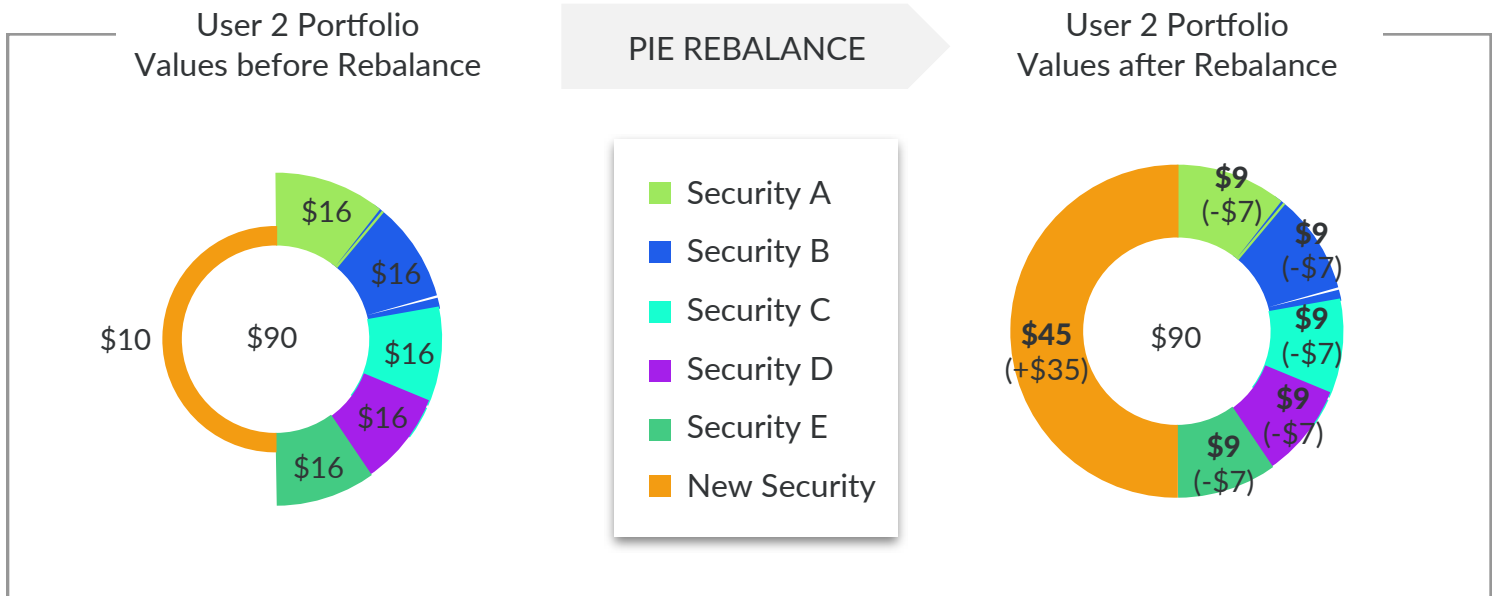
**Result:** The \$10 cash inflow will be used to push User 2's portfolio towards the new portfolio targets with a \$10 purchase.

3. User 2 makes cash withdrawal



**Result:** The \$20 cash withdrawal will be used to push User 2's portfolio towards the new portfolio targets, with sales of Security A, B, C, D and E.

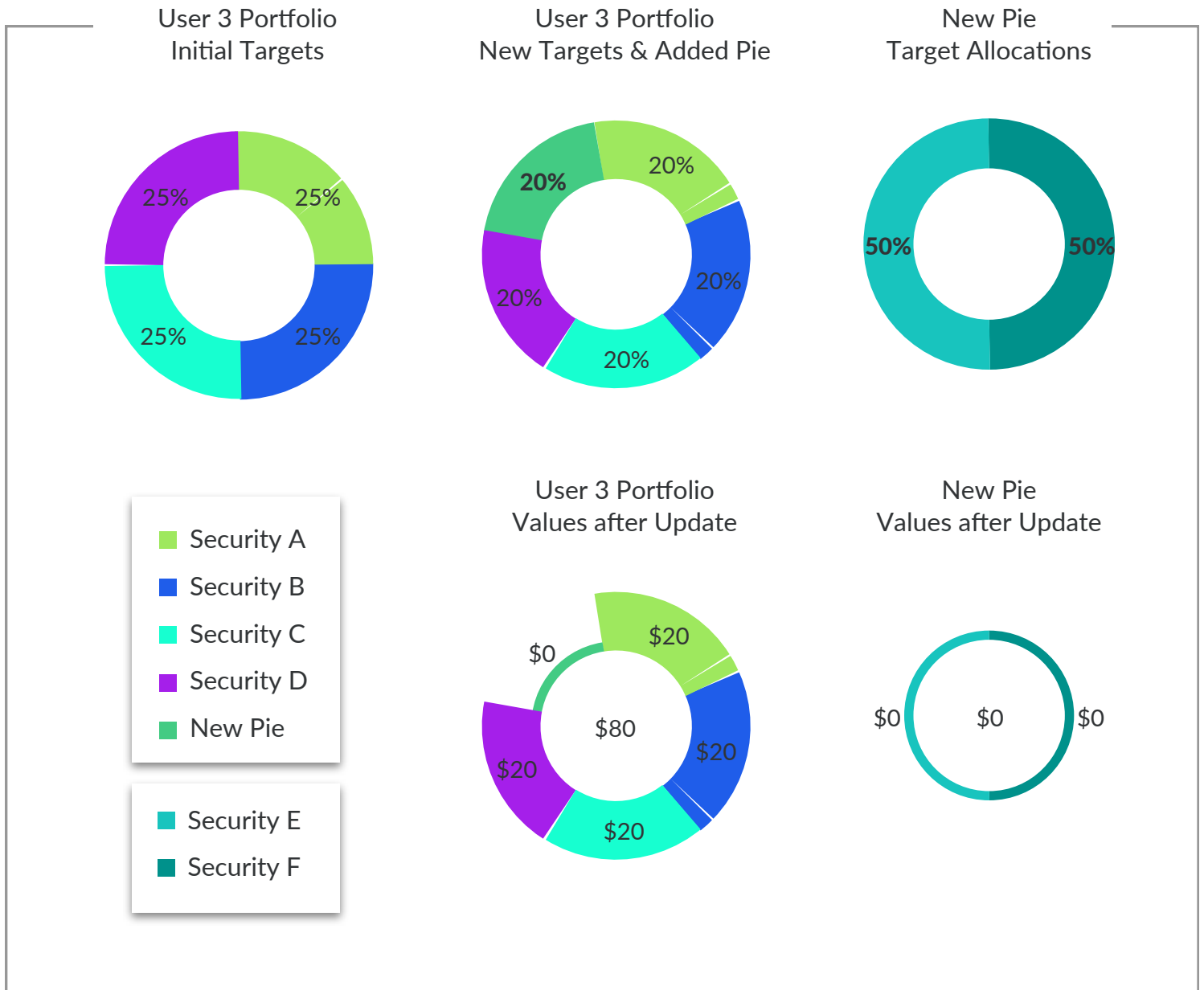
#### 4. User 2 makes rebalance



**Result:** The rebalance pushes User 2's portfolio values to the new target allocations.

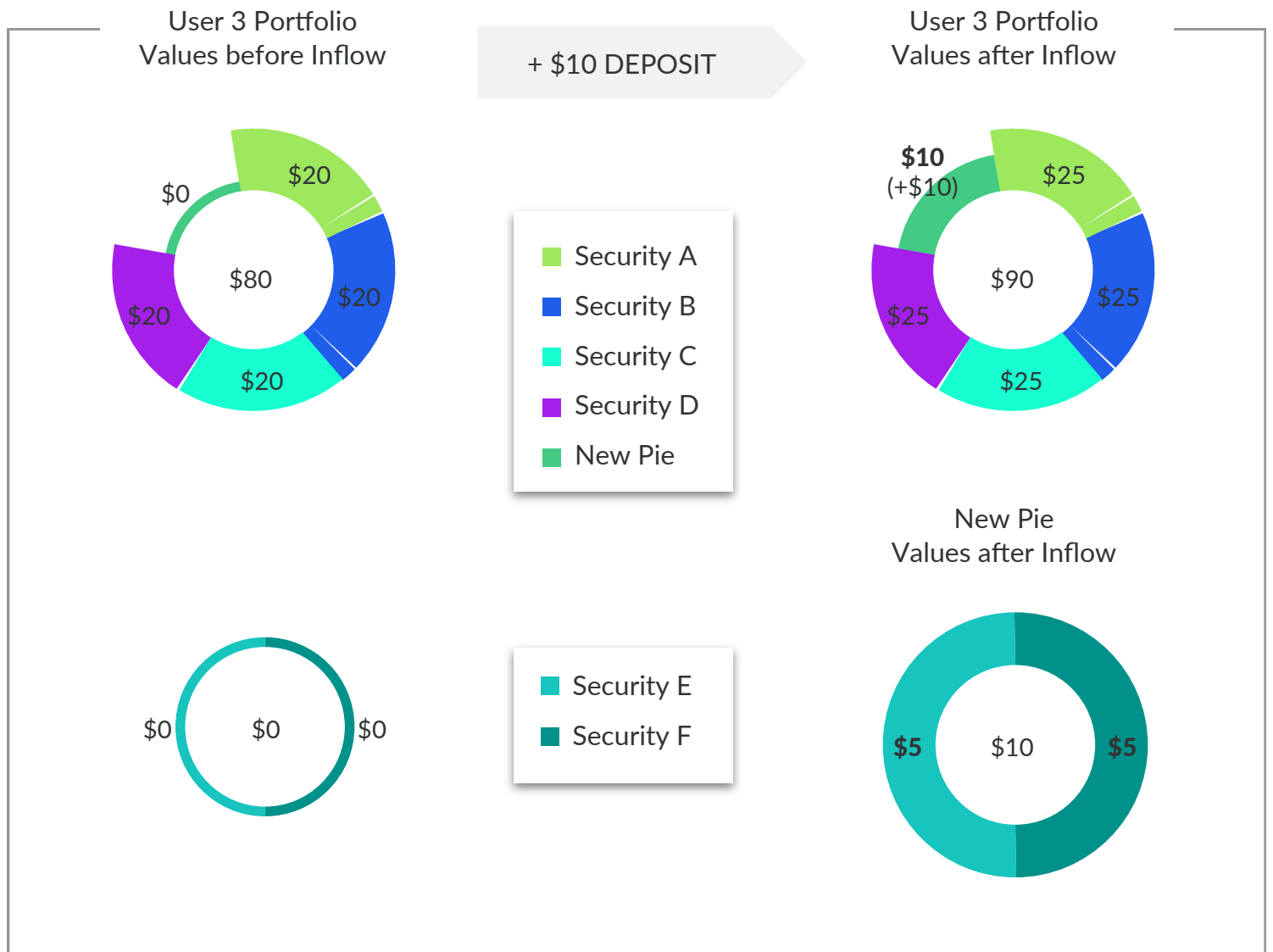
**Example 3 (Adding Slices):** User 3 starts with a portfolio with 4 equally weighted securities and then (1) adds a slice that references a Pie of 2 equally weighted securities. User 3 then (2) makes a deposit, (3) a withdrawal, and (4) eventually rebalances his portfolio.

1. User 3 add a New Pie and changes target allocations



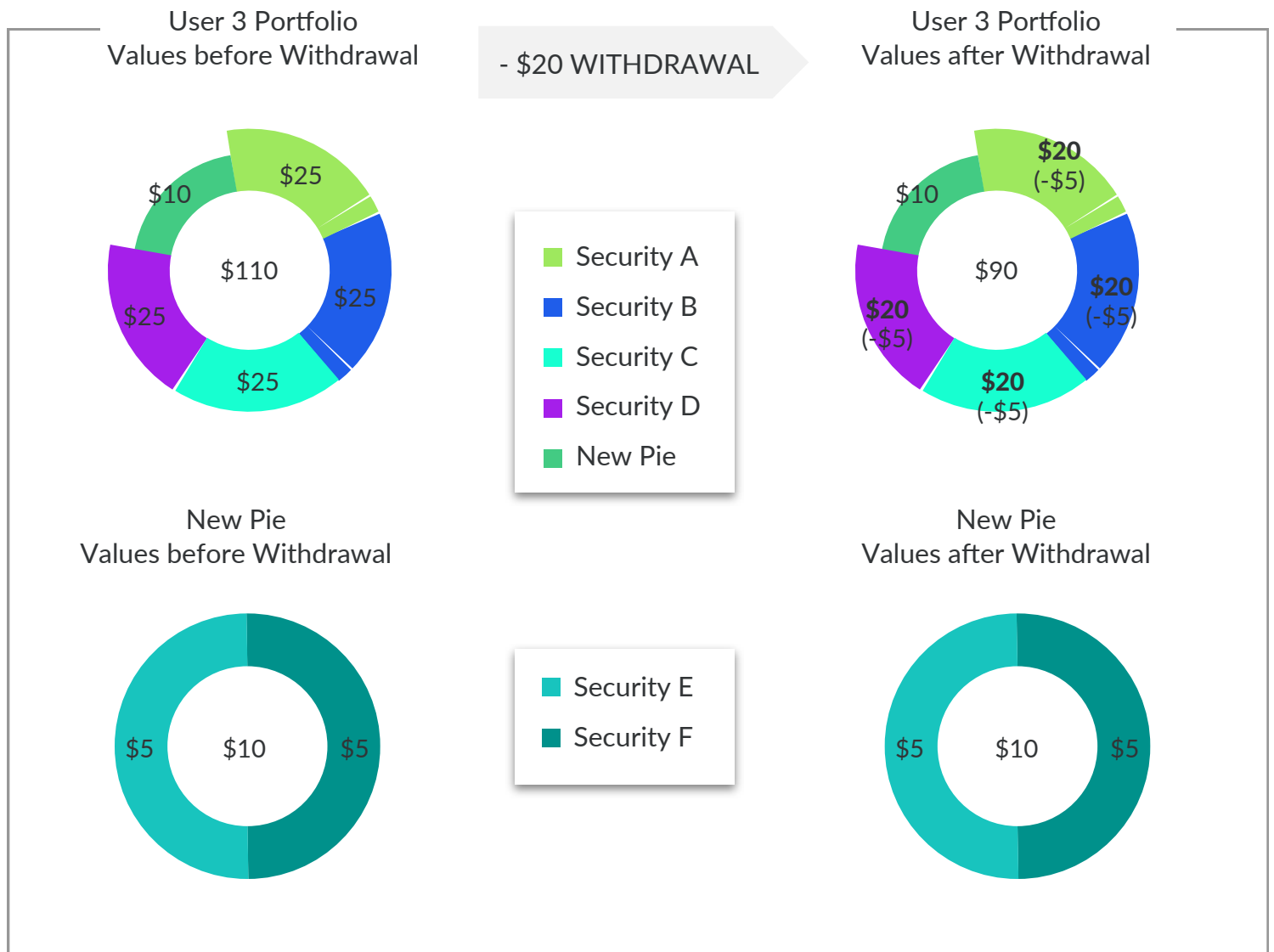
**Result:** Changing the target allocations and adding New Pie does not create any trading events. So New Pie will be unfunded until a trading event occurs.

2. User 3 makes a cash deposit



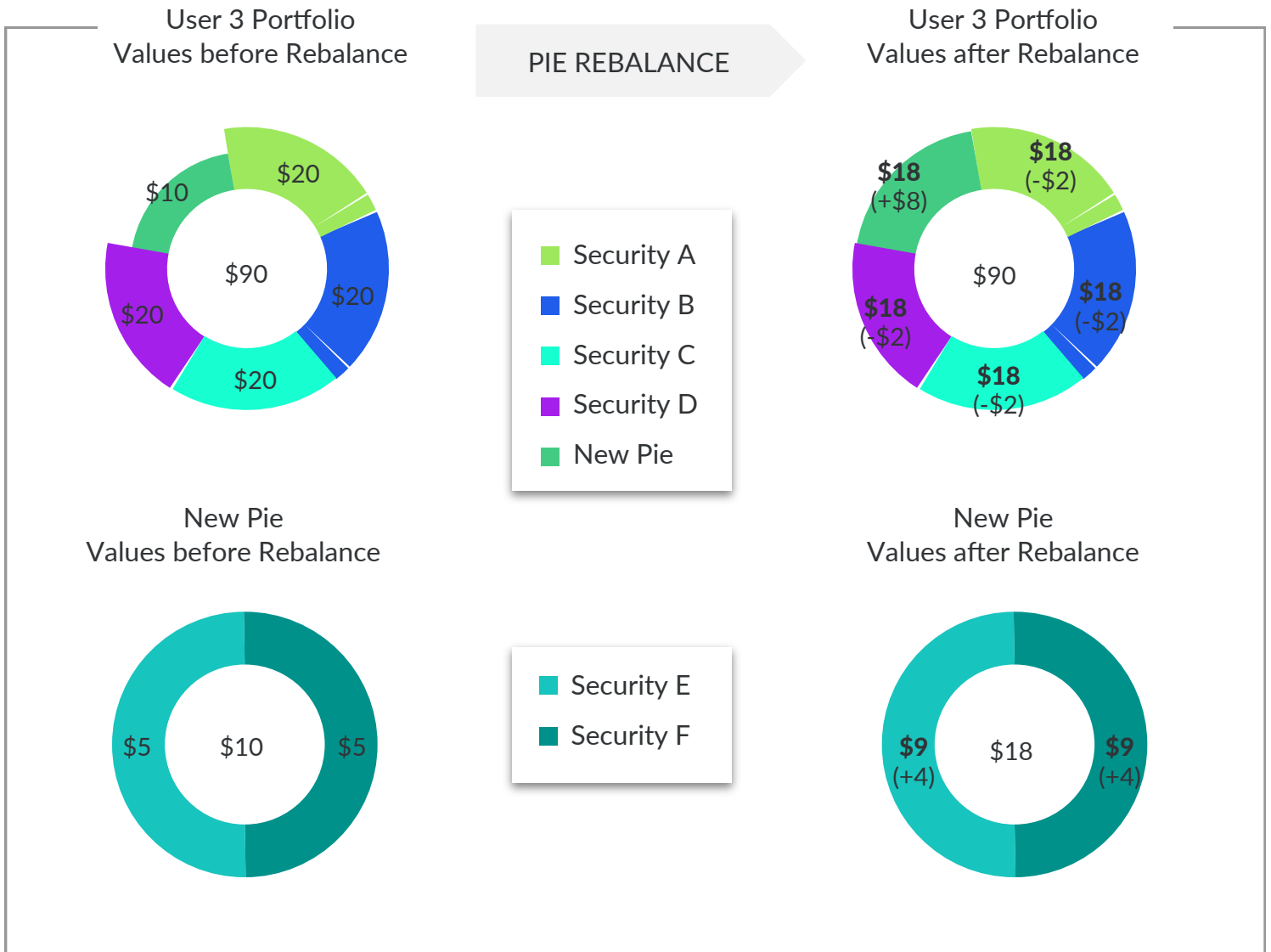
**Result:** The \$10 cash inflow will be used to push User 3's portfolio towards the new portfolio targets with a \$10 purchase flowing into the New Pie with a \$5 purchase of Security E and Security F.

3. User 3 makes a cash withdrawal



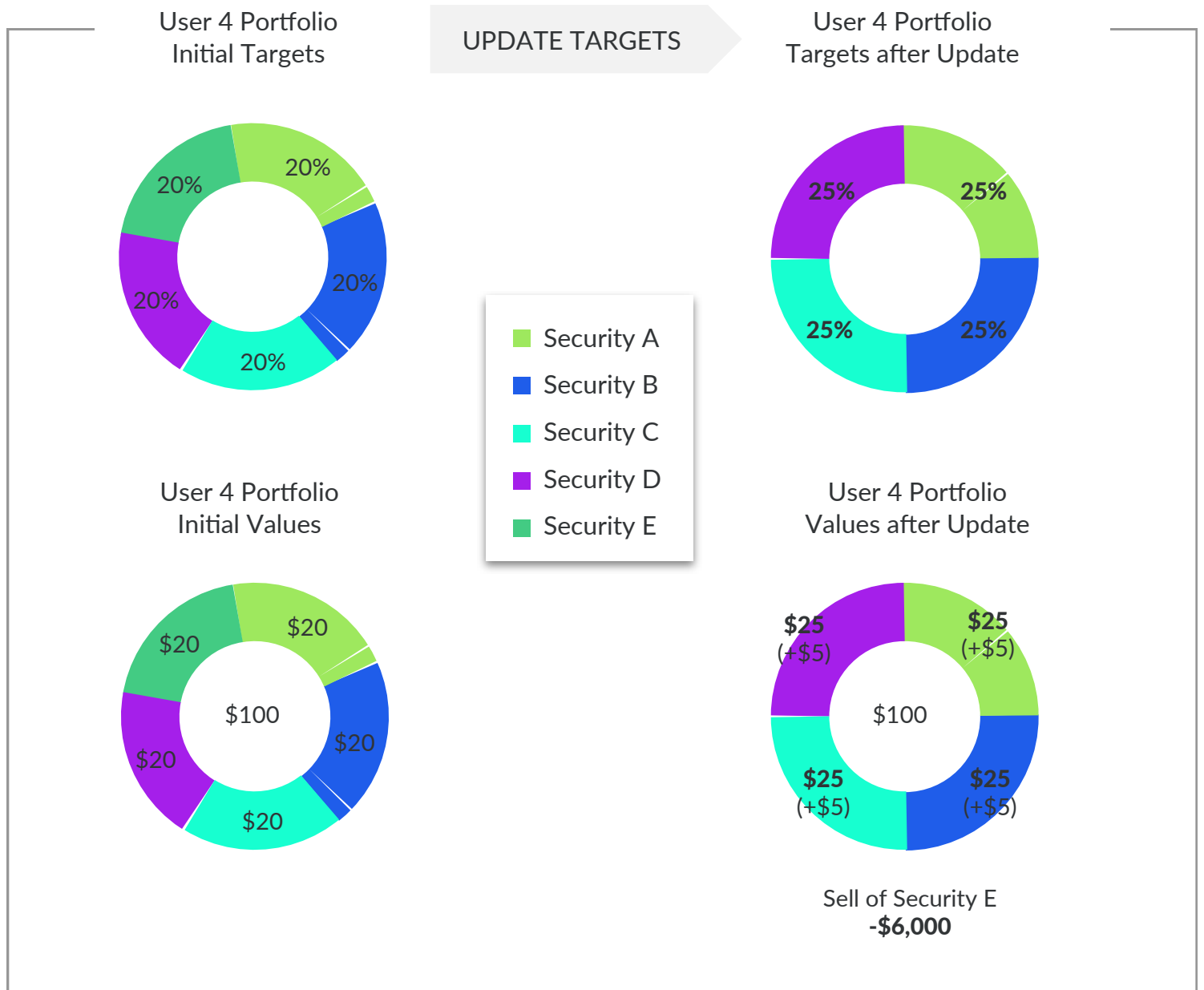
**Result:** The \$20 cash withdrawal will push User 3's portfolio towards the new portfolio targets, with sales of Securities A,B, C, and D.

#### 4. User 3 rebalances portfolio



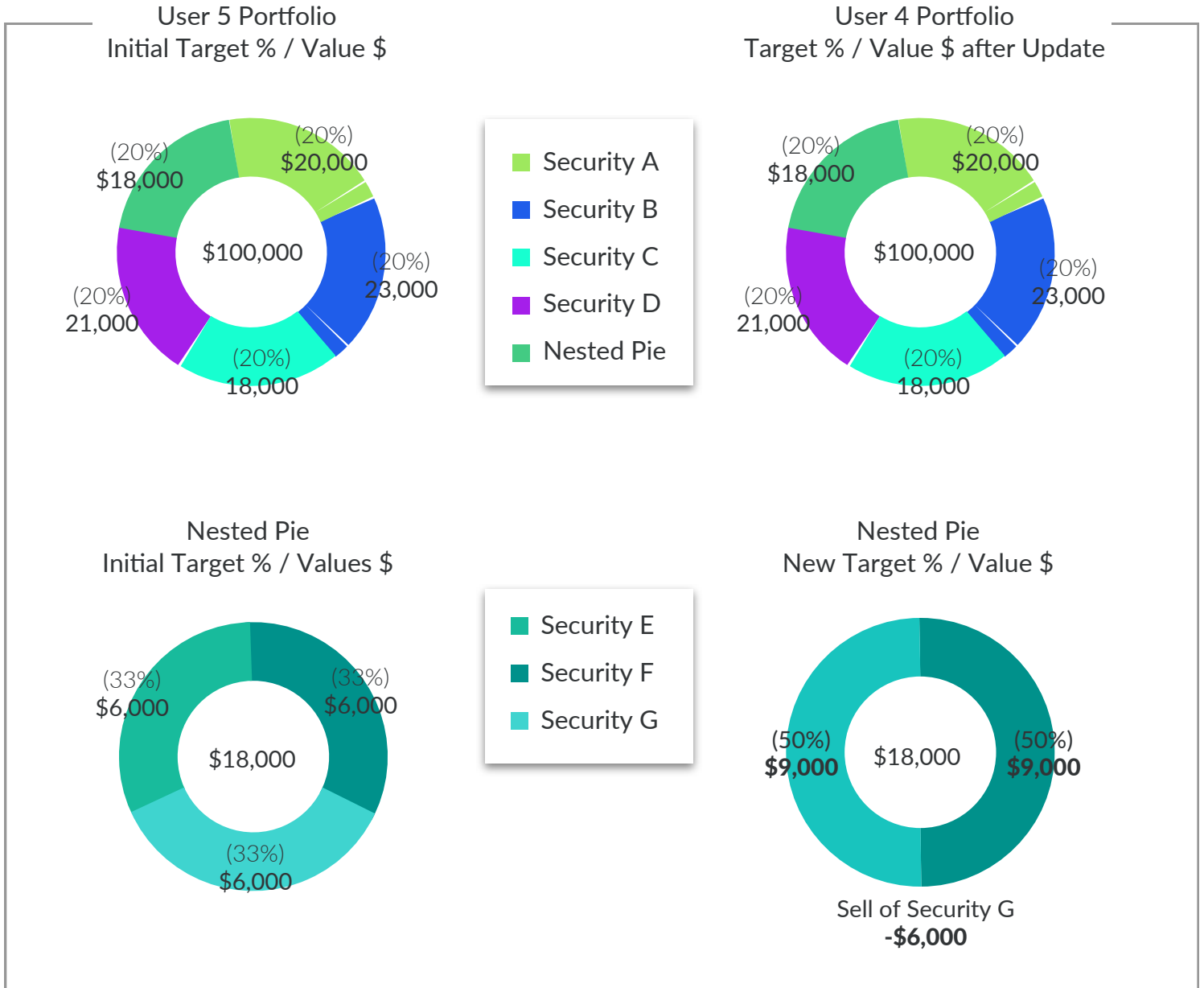
**Result:** The rebalance pushes User 3's portfolio values to the new target allocations.

**Example 4 (Liquidation):** User 4's portfolio initially contains 5 equally weighted securities. He removes one of the securities in his portfolio (liquidates) and sets equal targets for the 4 remaining securities.



**Result:** The update of target allocations causes immediate trading activity because of the liquidation of Security E. User 4 sold all of Security E and used the proceeds to purchase all other slices.

**Example 5 (Liquidation):** User 5's portfolio consists of five equally weighted slices and one of the slices is comprised of a Pie with three equally weighted securities. He updates his targets by removing one of the securities in the nested Pie.



**Result:** The update of target allocations does cause immediate trading activity because of the liquidation of Security G. Security G will be sold and the proceeds will be used to purchase slices in the Nested Pie. The Top Level Portfolio values will be unchanged.