

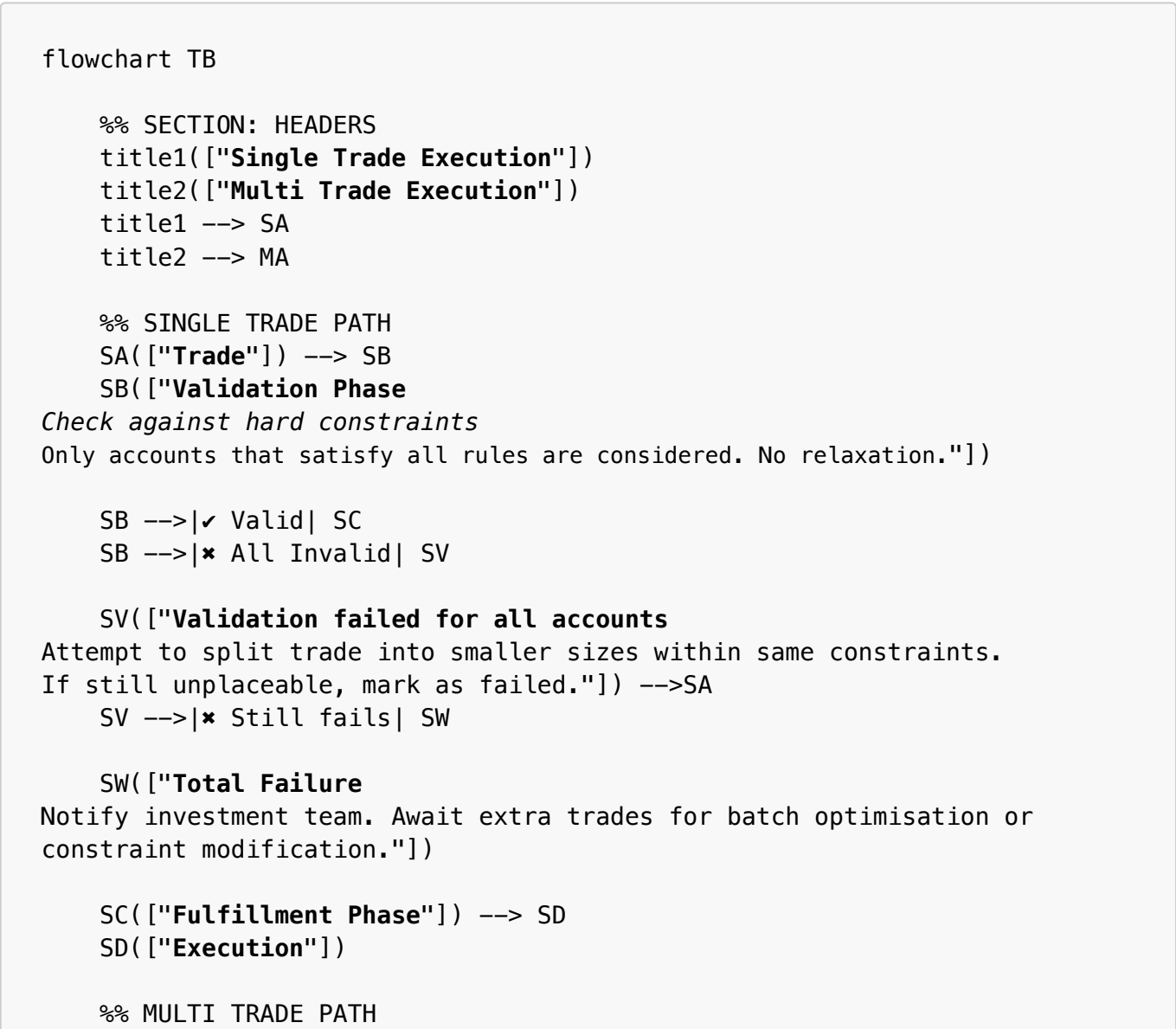
# Executive Summary

This plan outlines the implementation of an execution optimization framework for a hedge fund operating across multiple prime brokers. The primary objective is to develop and deploy a cost reduction algorithm that intelligently allocates trades to prime brokers, minimizing overall execution costs while respecting key constraints such as liquidity, risk limits, and regulatory requirements. By integrating advanced optimization techniques, the system will enhance decision-making in trade assignment, balancing cost efficiency with operational feasibility. Successful implementation will lead to significant reductions in trading expenses, improved execution quality, and a stronger competitive position for the hedge fund.

## Algorithm Overview: Validation and Fulfillment Stages

The diagram below provides a high-level overview of the execution optimization algorithm. It illustrates the two key stages of the process: the Validation Phase, where constraints and business rules are applied to ensure feasibility, and the Fulfillment Phase, where optimization priorities guide the selection of the best trade assignments. This structured approach ensures that all trade allocations comply with necessary restrictions while achieving cost-efficient execution across multiple prime brokers.

### Single trade execution



MA(["Batch of Trades"]) --> MB

MB(["Batch Validation Phase

*Each trade must meet hard constraints in each account, try combination of each account and trade while splitting trades progressively"])*

MB -->|✓ All Valid| MC

MB -->|△ Partial Valid| MP

MB -->|✖ All Failed| MV

MP(["Proceed with valid trade combinations. Choose which combination of trades from set of valid trade combinations best meets optimisation priorities."]) --> MC

MV(["All trades failed

No fallback possible without constraint changes"])

MW(["Total Failure

Notify investment team. Await extra trades for batch optimisation or constraint modification."])

MC(["Fulfillment Phase"]) --> MD

MD(["Execution"])

%% Shared Inputs

ACC(["Accounts Info"]) --> SE

ACC --> MF

SE(["Metrics (Hard Constraints)"]) --> SB

SE --> MB

MF(["Optimisation Inputs"]) --> SC

MF --> MC

%% Styles

style SA fill:#e3f2fd,color:#0d47a1,stroke:#0d47a1

style SB fill:#bbdefb,color:#0d47a1,stroke:#0d47a1

style SC fill:#90caf9,color:#0d47a1,stroke:#0d47a1

style SD fill:#64b5f6,color:#0d47a1,stroke:#0d47a1

style SV fill:#ffe0b2,color:#e65100,stroke:#e65100

style SW fill:#ffab91,color:#bf360c,stroke:#bf360c

style MA fill:#e1f5fe,color:#01579b,stroke:#01579b

style MB fill:#b3e5fc,color:#01579b,stroke:#01579b

style MC fill:#81d4fa,color:#01579b,stroke:#01579b

style MD fill:#4fc3f7,color:#01579b,stroke:#01579b

style MP fill:#fff9c4,color:#f57f17,stroke:#f57f17

style MV fill:#ffccd2,color:#b71c1c,stroke:#b71c1c

style MW fill:#ff8a65,color:#bf360c,stroke:#bf360c

style SE fill:#f0f4c3,color:#827717,stroke:#827717

style MF fill:#fce4ec,color:#880e4f,stroke:#880e4f

style ACC fill:#fff3e0,color:#bf360c,stroke:#bf360c

style title1 fill:#f5f5f5,stroke:none,color:#1a237e

style title2 fill:#f5f5f5,stroke:none,color:#01579b

## Execution Framework Overview

This model outlines the decision-making process for routing bond trades into appropriate Prime Brokerage (PB) accounts. It is split into two main paths based on the volume of trades: **Single Trade Execution** and **Multi Trade Batch Execution**. Each path aims to ensure account-level constraint compliance while optimizing for operational or financial priorities like financing cost and constraint headroom.

### Single Trade Execution

In this path, each trade is processed independently. The process follows a strict validation-first model, where the trade is only considered for accounts that fully satisfy all applicable hard constraints.

- **Validation Phase:** The trade is tested against all accounts. Only accounts where post-trade metric values remain within predefined constraints are considered.
- **Failure Handling:** If no account satisfies the constraints, the trade is split into smaller sizes and re-tested. If still invalid, the trade is marked as a *total failure* and flagged for manual intervention.
- **Fulfillment Phase:** If one or more valid accounts exist, the best one is selected (e.g., lowest cost or best optimization score), and the trade is executed.

### Multi Trade Batch Execution

When multiple trades are available, the system leverages the opportunity to apply **batch-level optimization**. This enables more efficient use of constraint capacity and better overall performance on optimization priorities.

- **Batch Validation Phase:** Trades are evaluated in combinations, including partial splits across accounts. Each combination is tested against account-level constraints to form a pool of valid trade-account pairings.
- **Partial Success Handling:** If only a subset of trades pass validation, the system identifies the *most optimal valid subset* (e.g., maximizing constraint headroom or minimizing financing cost) and proceeds with those.
- **Total Failure:** If no valid trade combinations exist, the batch is halted, and the trades are flagged for review or rerun with modified constraints.
- **Fulfillment Phase:** The valid trade-account assignments are executed, and metrics are updated accordingly.

### Shared Data & Logic

Both paths use a shared pool of data, including:

- **Account Information:** Starting positions, constraints, and metric definitions.
- **Metrics Engine:** Calculates each account's post-trade metrics using weighted value formulas.
- **Optimization Inputs:** Weights provided by the investment team (e.g., for cost, margin efficiency) are used in scoring and selection logic.

### Design Philosophy

The system prioritizes constraint satisfaction — no trades are allowed to breach hard limits. Optimization is applied only *after* all constraints are satisfied. This two-stage design ensures regulatory and risk

compliance while enabling smart allocation of trades across PB accounts to minimize cost and maximize operational efficiency.

```

flowchart TD
    %% Inputs
    A1["Accounts
    • trades
    • cash
    • metrics: 1 <= k <= 20"]
    A2["Portfolio
    • accounts"]
    A3["Trades
    • side: buy/sell
    • security name
    • type: gov or corp
    • quantity
    • price
    • accrued interest"]

    %% Derived metrics
    M["Metric Calculations
    • metric value
    • constraint (optional)
    • constraint type
    • optimization priority"]

    %% Optimization context
    OP["Optimization Priorities
    • description
    • weight"]

    %% Engine
    E["Optimization Engine
    • Apply hard constraints
    • Score priorities
    • Optimize assignment"]

    %% Output
    R["Trade to Account Assignment"]

    %% Arrows and Flow
    A1 --> M
    A2 --> A1
    A3 --> M
    M --> E
    OP --> E
    E --> R

    %% Styling
    style A1 fill:#e0f7fa,stroke:#00796b,color:#004d40
    style A2 fill:#f1f8e9,stroke:#689f38,color:#33691e
    style A3 fill:#fce4ec,stroke:#d81b60,color:#880e4f
  
```

```
style M fill:#fff3e0,stroke:#fb8c00,color:#e65100
style OP fill:#ede7f6,stroke:#5e35b1,color:#311b92
style E fill:#e3f2fd,stroke:#1976d2,color:#0d47a1
style R fill:#dcedc8,stroke:#558b2f,color:#33691e
```