

Real Time Alerting System

Algorithm and Data Structures

The system would be based on a stream processing model, where transactions are processed as they arrive.

The data structures used would be:

- **Priority Queue:** To keep track of transactions exceeding a pre-defined threshold. It keeps track of transactions within a certain time window (default is 3600 seconds or 1 hour). It uses a deque to efficiently add and remove transactions.
- **Sliding Window:** To monitor the number of transactions in a given time frame. It keeps track of transactions based on their amount. It uses a heap to efficiently add transactions and retrieve them in order of their amount. Note that Python's `heapq` module provides a min-heap, so we store the negative amount to get a max-heap behavior.

Scalability, Data Integrity, and Fault Tolerance

- **Scalability:** The system can be scaled horizontally by partitioning the transaction stream based on fund managers or syndicates. Each partition can be processed independently.
- **Data Integrity:** Transactions are immutable once they are in the system, ensuring data integrity. Any necessary corrections must be made through compensating transactions.
- **Fault Tolerance:** The system can be made fault-tolerant using techniques like replication (keeping copies of the data) and checkpointing (saving the state of the system at regular intervals).