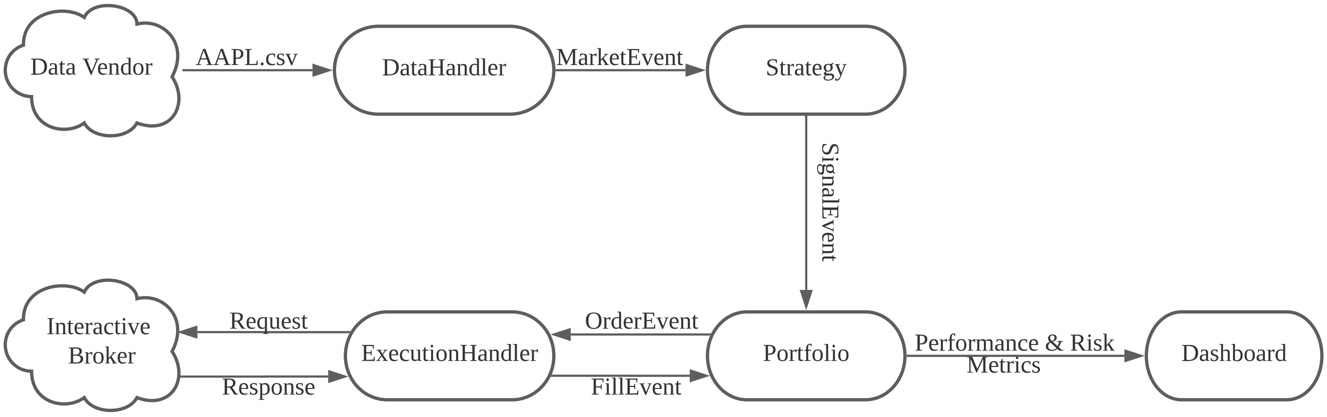
Project Proposal

Chuang Yao, Jianqi Zhu, Kongjun Qin.

We will be developing a basic event-driven backtesting library (HackTest) as the course project for OOP I. HackTest will be a library inspired by a series of articles on QuantStart about event-driven backtesting with Python. It will be written in C++ with the following in mind:

- object-oriented approach & code reusability
- supporting different order types
- realistic emulation of market transactions
- capable of handling real-time market data
- live metrics of performance and risk management

The library will likely function as illustrated bellow:



For now, we will focus on backtesting with historical data. We assume price data have already been downloaded from data vendors as CSV files. However, this framework should be easily adapted to a live trading environment.

Based on the downloaded market data, `DataHandler` will emit a `MarketEvent` object to the `Strategy` object.The `Strategy` class is implemented by library users and can be customized to support different trading strategies, from momentum-based ones to machine learning. Upon receiving the `MarketEvent` , it will process the data and emit a `SignalEvent` to `Portfolio` with instructions on how to adjust current positions. Then `Portfolio` will send an `OrderEvent` to `ExecutionHandler` on what to buy and what to sell. `ExecutionHandler` will talk to an actual or a simulated interactive broker. `FillEvent` contains feedback from the broker to the `Portfolio` about the details of orders executed, such as price filled and transaction costs. The `Dashboard` will be updated with real-time performance and risk management metrics.