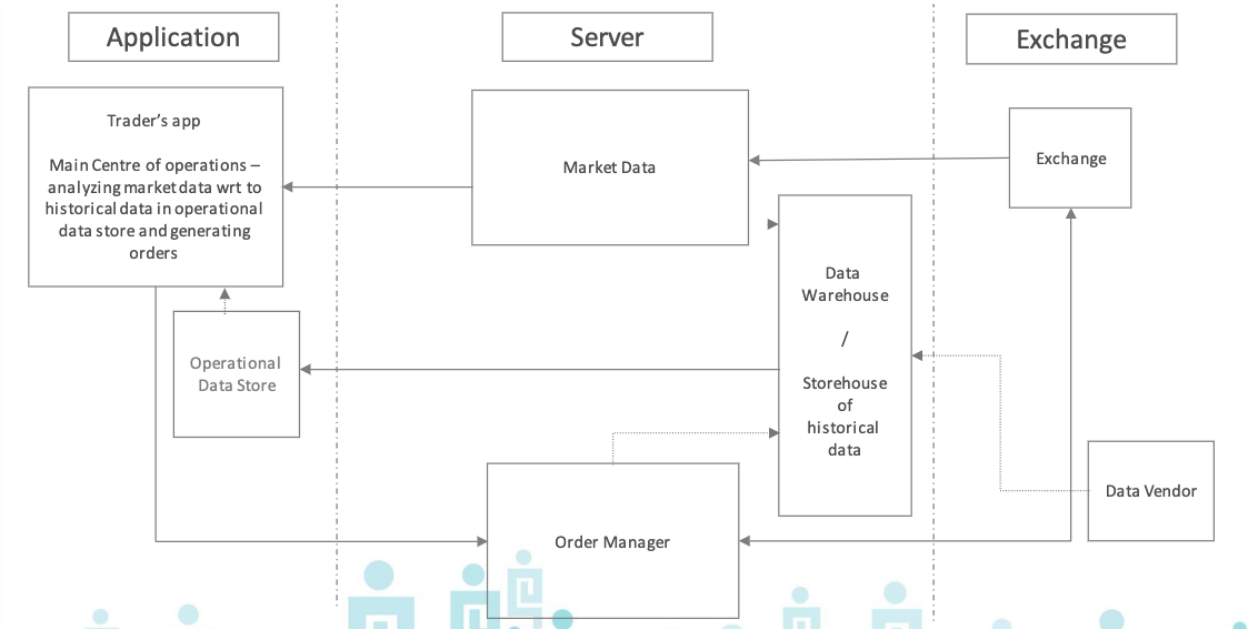
TIO01 Lecture Summary

Traditional trading system

The traditional trading system consists of

- A system to read data from the market
- A storehouse of historical data
- A tool to analyze historical data
- A system where the trader can input his trading decisions
- A system to route orders to the exchange



Overview:

The system could be broken down into three components:

- **Exchange:** The exchange (and other data sources)
- **Server:** The server is a central node for handling market data and forwarding it to multiple trading applications. It also relays data to data storage. Additionally, it also performs order management.
- **Applications:** The applications in the trader's machine which performs all the processing.

Data Flow and functions of the Processes:

Market Data: Market data is a system to read data from the exchange (market data adaptor)

Data Warehouse: Data Warehouse is a storehouse of historical data (which could also be procured from third-party vendors)

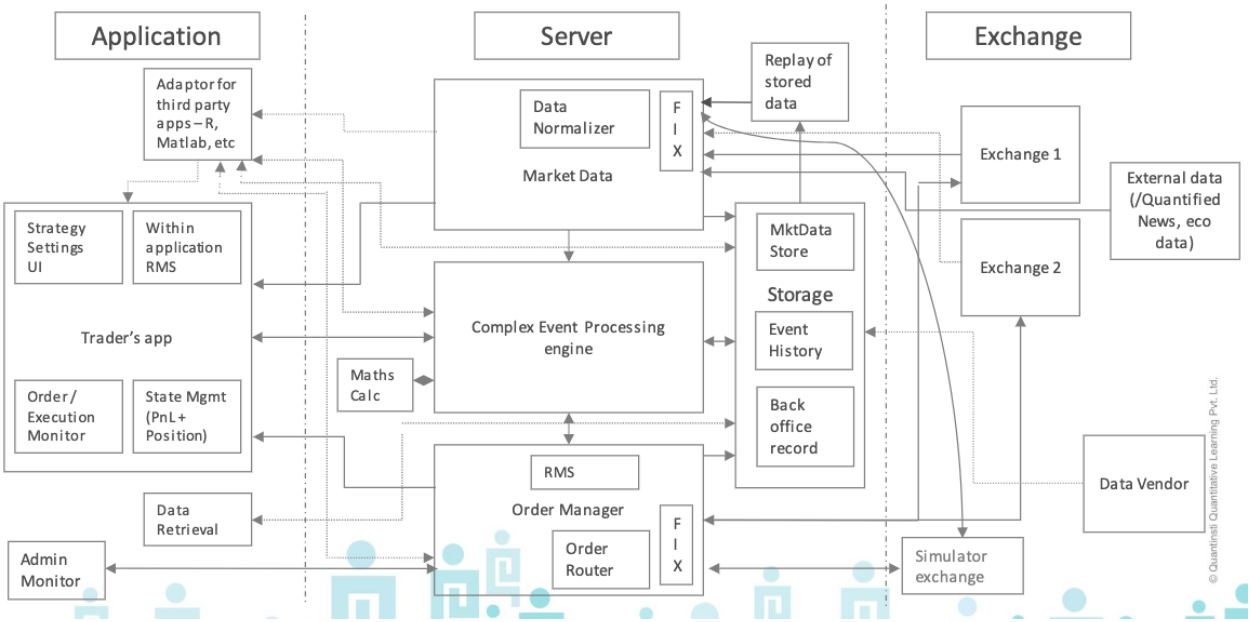
Operational Data Store: Operational Data Store is a subset of the information stored locally for trading app use.

Trader's app: The trader's app analyses current data against patterns discovered in the operational data store. It also generates the orders and forwards them to the order manager.

The order manager: The order manager would then route the orders to the exchange.
The order manager also gets a response from the exchange about order executions.

The data warehouse: The data warehouse stores the log of orders and their execution details. It also stores the historical data of the tickers.

The transformation from a Traditional system to an Automated system



Complex Event Processing Engine(CEP): The CEP handles the automation of monitoring of prices, trading decisions, and execution of orders. It gets feedback about orders, current positions and executions. It uses the same information for making trading decisions.

Maths Calculation: Complex mathematical operations are handled in a dedicated maths calculation block in the server block. Maths calculations block handles the calculations such as (not limited to) options greeks calculation, statistical parameters, technical indicators etc.

Transformation of Application Layer to Server Layer: The role of the application layer has reduced drastically w.r.t. traditional trading system :

- The order management part has moved to "Server → Order Manager".
- Few functions of risk management have moved to "Server → Order Manager → RMS". And RMS is now automated and checked by the OMS(Order Management System) before an order is routed to exchanges.

Transformation of Exchange Layer: Exchange layer now accommodates interfaces with multiple exchanges. FIX protocol is added in the market data block and order manager block.

Changes in Order Manager: FIX protocol is added in the market data block and order manager block. An order router is added to the OMS to route orders from the same OMS to multiple exchanges.

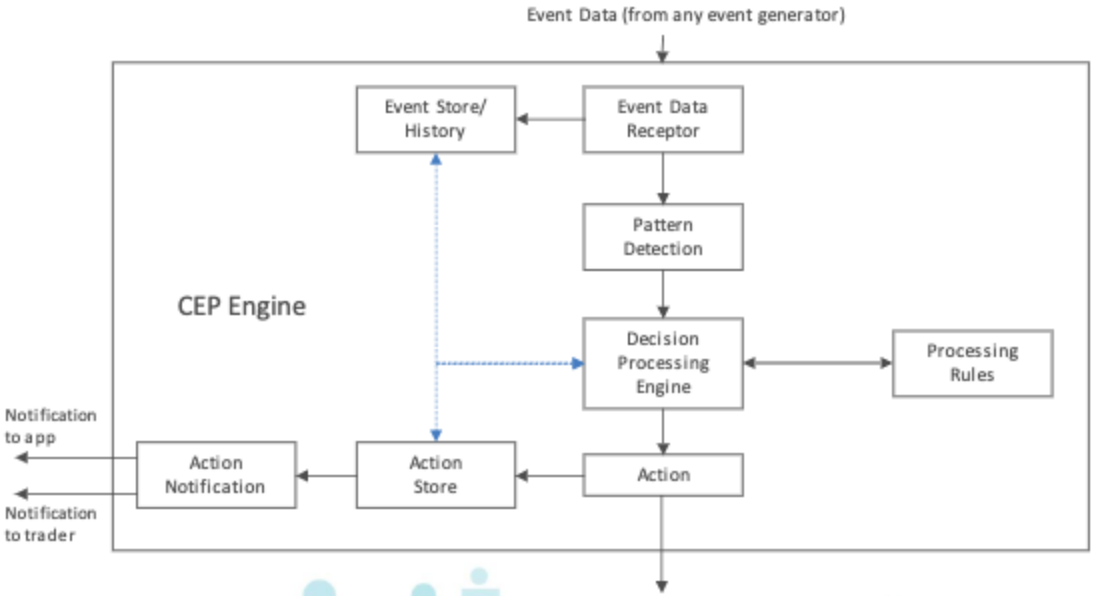
Changes in Market Data Block: Addition of data normalizer block in the market data adaptors to convert data from multiple exchanges into a standard format. With the increase in complexity of the data, the sophistication of the data tools has gone up. Third-party data analytical applications have to be tightly integrated with all the blocks.

Changes in Storage Block: Regulatory requirements have complicated storage requirements – requiring storage of trade information (in addition to market data that firms were storing for in house use).

Changes in CEP Block: The CEP engine has its own storage facility of event history to identify future opportunities (without doing entire re-calculations). Inputs to it could also be diverse and not just limited to exchange market data. These could be quantified news scores, or economic/earnings data in a standardized format

Simulator Exchange: To validate the correctness of the implementation of an algorithmic trading strategy, simulators were added which would simulate the behaviour of exchanges on receiving market data.

CEP Engine



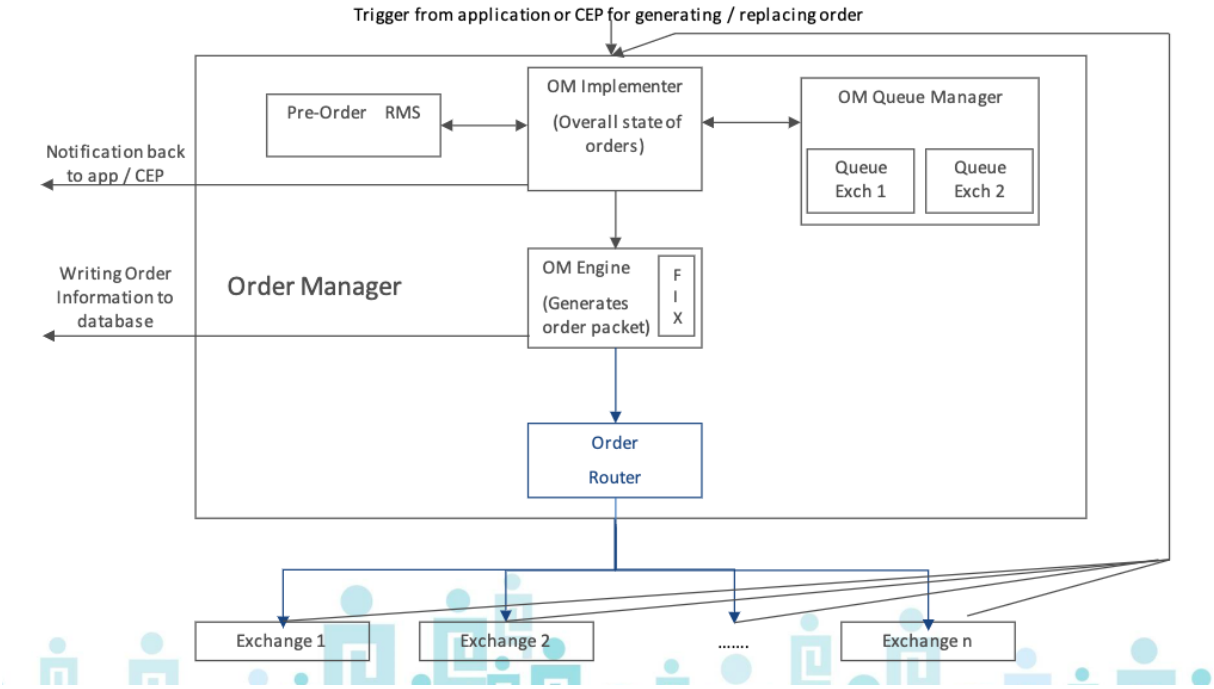
Functions of CEP: The CEP is the heart of the system, which listens to market data and determines what actions to take based on algorithm settings.

- The first task is to decode the event data in a receptor, and store it in event history.
- The next block checks the event for known patterns.
- Based on Decision Processing Rules, a Decision Processing Engine determines what to do in case a pattern is recognized.
- If the Decision Processing Engine determines that an action has to be done, then it is conveyed to the action destination outside the CEP. For regulatory and for analysis purposes, this information is stored in an action store.

Inputs to the CEP: The input to the CEP is 'Event Data'. This could be (i) market data or (ii) change of strategy settings in the application. This could also be (iii) order reports (execution or acknowledgement) from the exchange.

Outputs of the CEP: The output of the CEP module is (i) an action (sending an order request to the order manager) and (ii) sending notifications to both the trader and the application.

Order Manager



Functions of Order Manager: The order manager generates and manages orders sent from the system to multiple destinations. It also performs RMS in real-time before sending an order. The RMS checks max order size, net portfolio position, max trade value, etc. and checks the OM Queue for each destination. If the queue is free, then OM prepares a packet of the orders to send to the respective exchange. For FIX protocol destinations, the orders are generated in FIX format.

Inputs to the Order Manager: The input to the OM is the signal from the CEP block. Another input to the OM is order acknowledgements and execution reports from the exchanges.

Outputs to the Order Manager: The output of the OM is

- orders routed to exchanges/other destinations
- notifications back to the application
- writing order information into a database in the Storage block