E.M.T.E

Exchange Matching and Trading Engine

Made by Massimo Di Pierro

What's in the box?

- A matching engine with support for market orders (buy/sell), limit orders, and stop orders
- A fully web-based trading platform to submit oders, check quotes and positions

Why?

- Scalable
- Uses HTML5 WebSocket for information delivery (requires HTML5 support)
- Strong on security
- Cross-platform - fully in browser
- Ideal for teaching
- A proof of concept of modern design

Technology?

- Implemented in Python
- Uses web2py for the web interface
- Uses the Tornado web server + HTML5 WebSockets for Asynchronous communications (very scalable)
- Works with multiple databases: SQLite, PostgreSQL, MySQL, MSSQL, FireBird, Oracle, IBM DB2, Informix, Ingres

[E]xchange [M]atching and [T]trading [E]ngine

Welcome Massimo [logout | profile | password]

full-stack exchange and trading plaform (fast, scalable, secure)

your order 1:sell intc 2000@200.0/0 was submitted

Index | Products | P&L | About

Product: intc

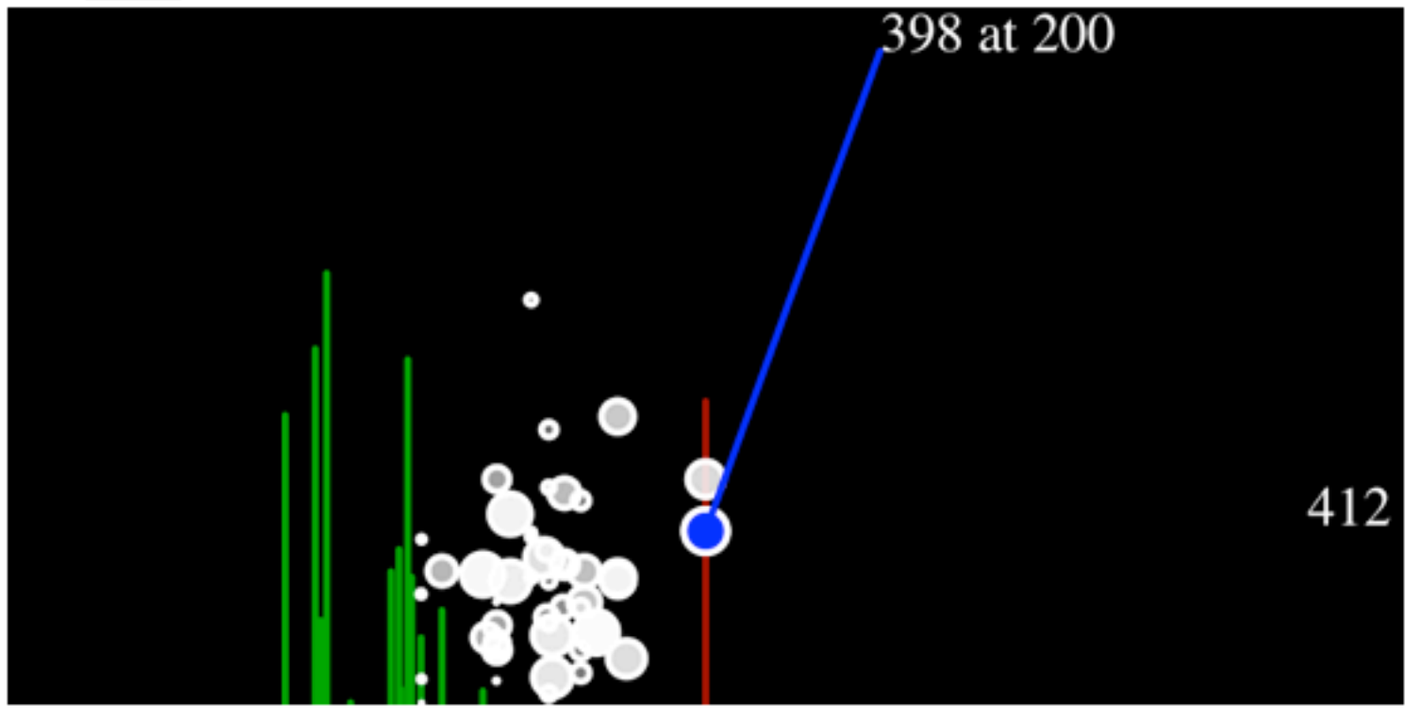
Buy Sell: buy

Quantity:

Price: 0 0 for market order

Stop Price: 0 0 if not stop order

Submit



Copyright © 2010

[E]xchange [M]atching and [T]trading [E]ngine

Welcome Massimo [logout | profile | password]

full-stack exchange and trading plaform (fast, scalable, secure)

your order 1:sell intc 2000@200.0/0 was submitted

Index | Products | P&L | About

Product: intc

Buy Sell: buy

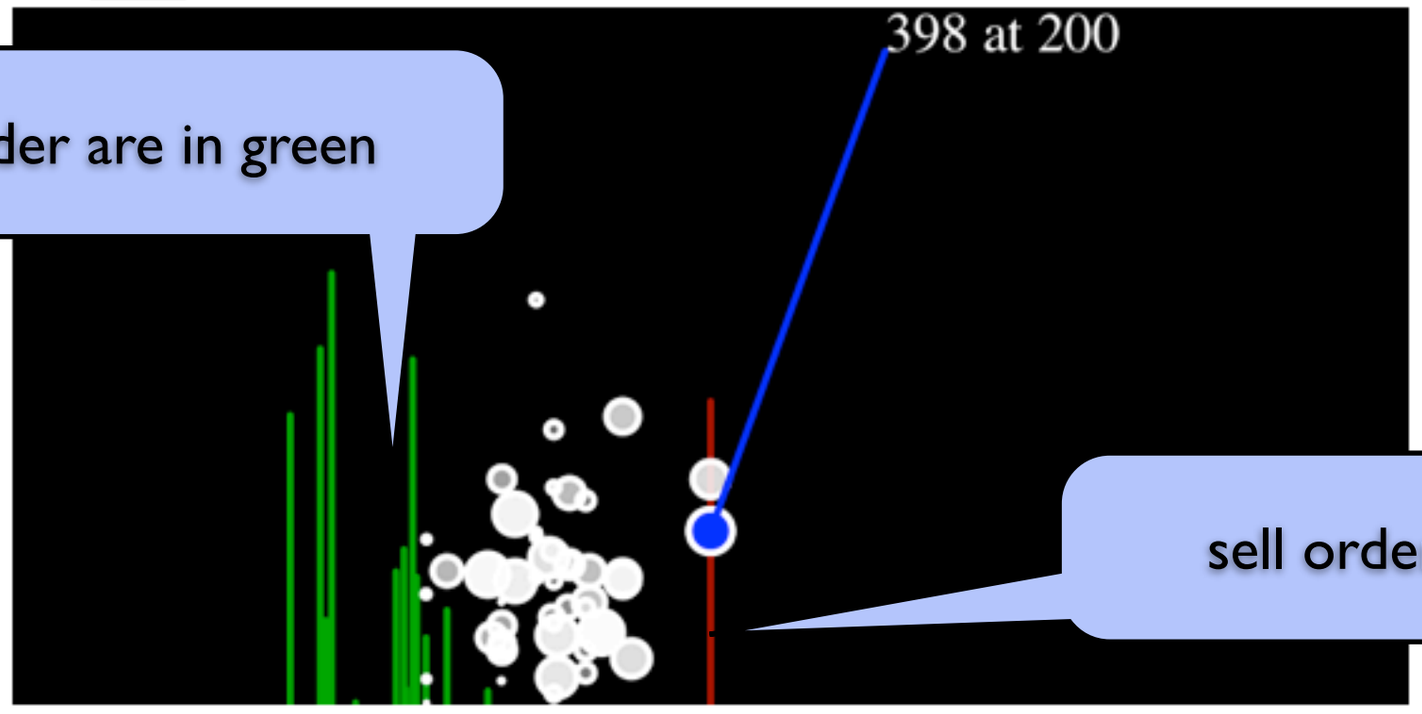
Quantity:

Price: 0 — 0 for market order

Stop Price: 0 — 0 if not stop order

Submit

submit an order



Copyright © 2010

[E]xchange [M]atching and [T]trading [E]ngine

Welcome Massimo [logout | profile | password]

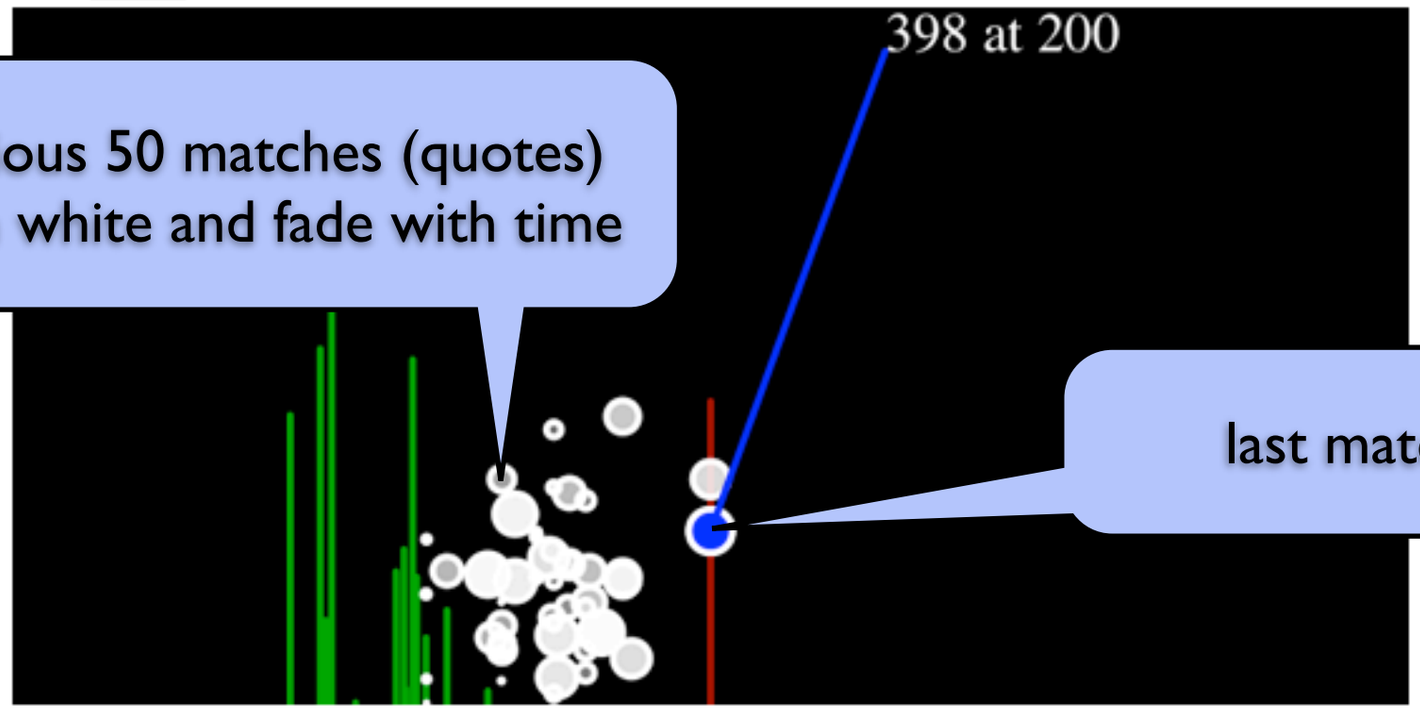
full-stack exchange and trading plaform (fast, scalable, secure)

your order 1:sell intc 2000@200.0/0 was submitted

Index | Products | P&L | About

Product: intc

Buy Sell: buy
Quantity:
Price: 0 — 0 for market order
Stop Price: 0 — 0 if not stop order
Submit



flash: last submitted order

previous 50 matches (quotes) are in white and fade with time

last match is in blue

Copyright © 2010

Robot Trader

The Robot Trader keeps the market busy

```
order #76 from 0:sell 342@0 (0.005603seconds)
order #77 from 0:sell 135@0 (0.004820seconds)
order #78 from 0:buy 153@137.27 (0.004469seconds)
order #79 from 0:sell 517@0 (0.004471seconds)
order #80 from 0:sell 306@0 (0.004367seconds)
order #81 from 0:buy 522@124.45 (0.004713seconds)
order #82 from 0:sell 806@159.54 (0.004944seconds)
order #83 from 0:sell 459@0 (0.004488seconds)
order #84 from 0:buy 790@0 (0.004931seconds)
order #85 from 0:buy 895@0 (0.005003seconds)
order #86 from 0:buy 965@0 (0.008467seconds)
order #87 from 0:buy 720@143.95 (0.007372seconds)
order #88 from 0:buy 556@156.08 (0.007781seconds)
order #89 from 0:buy 517@0 (0.004263seconds)
order #90 from 0:sell 539@165.23 (0.004558seconds)
order #91 from 0:sell 946@174.78 (0.004620seconds)
order #92 from 0:sell 291@0 (0.009321seconds)
order #94 from 0:sell 270@168.69 (0.005025seconds)
order #95 from 0:buy 105@185.92 (0.006063seconds)
order #96 from 0:sell 336@151.37 (0.005756seconds)
order #97 from 0:buy 329@136.16 (0.005425seconds)
order #98 from 0:sell 62@0 (0.005139seconds)
order #99 from 0:sell 441@0 (0.004919seconds)
order #100 from 0:sell 735@0 (0.006116seconds)
order #101 from 0:buy 563@0 (0.004453seconds)
```

Design

client
with
browser

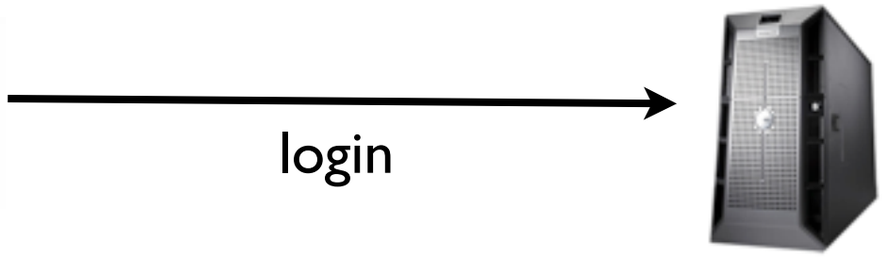
register

web server

matching engine

database

Design



Design

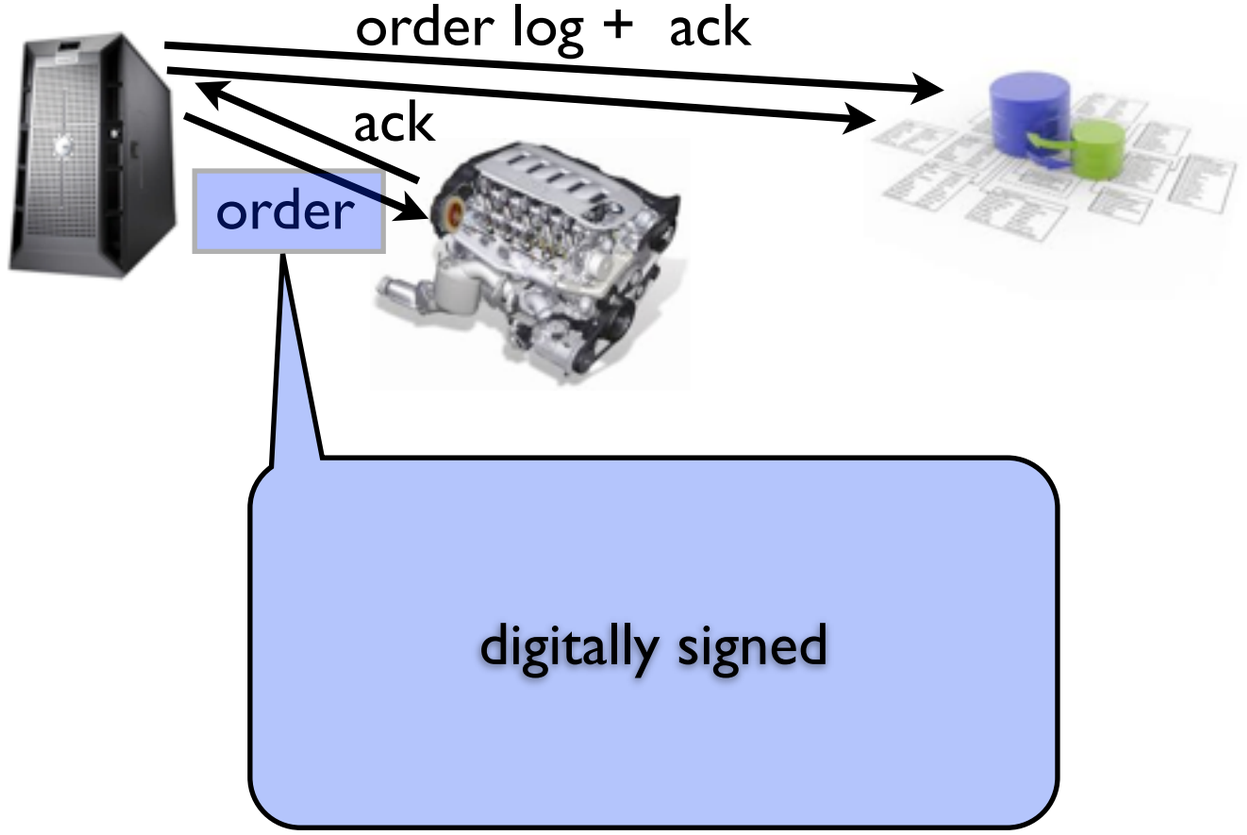
select product to trade

Design

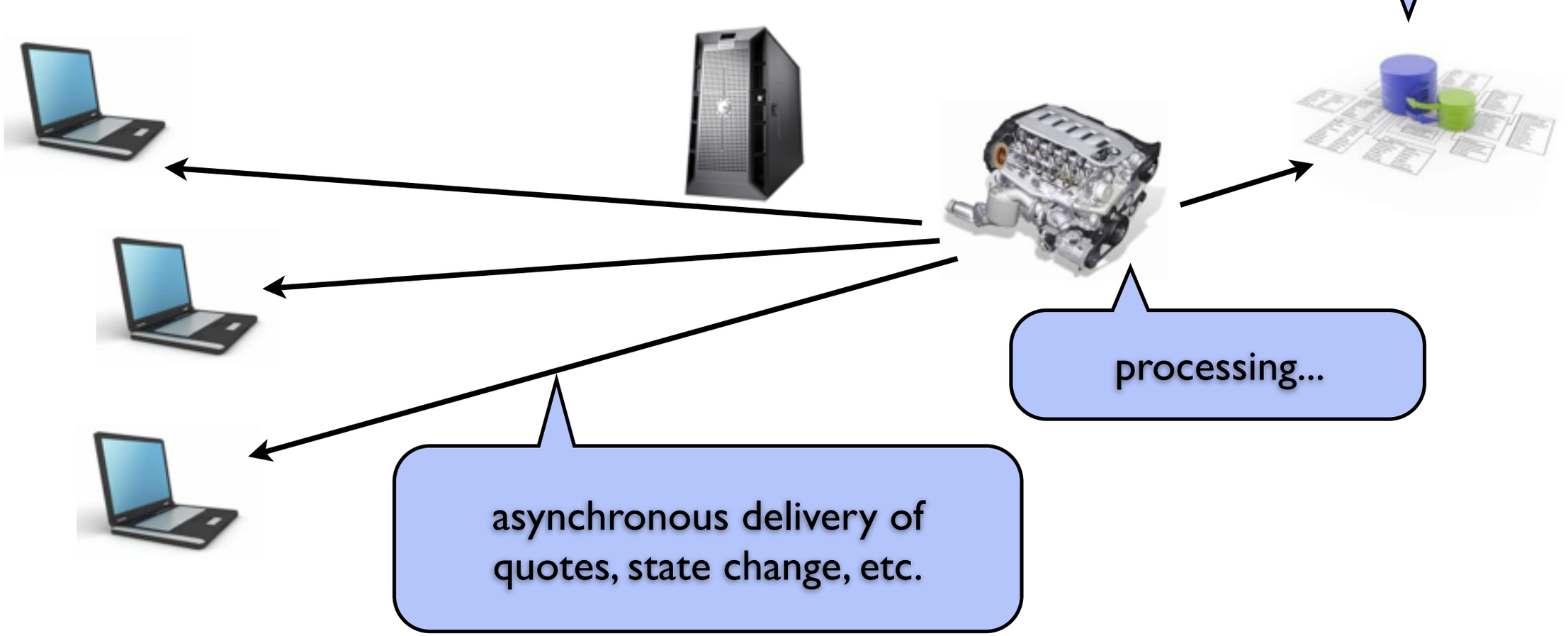
submit order (buy 1@35)

cookie based authentication
optional ssl encryption

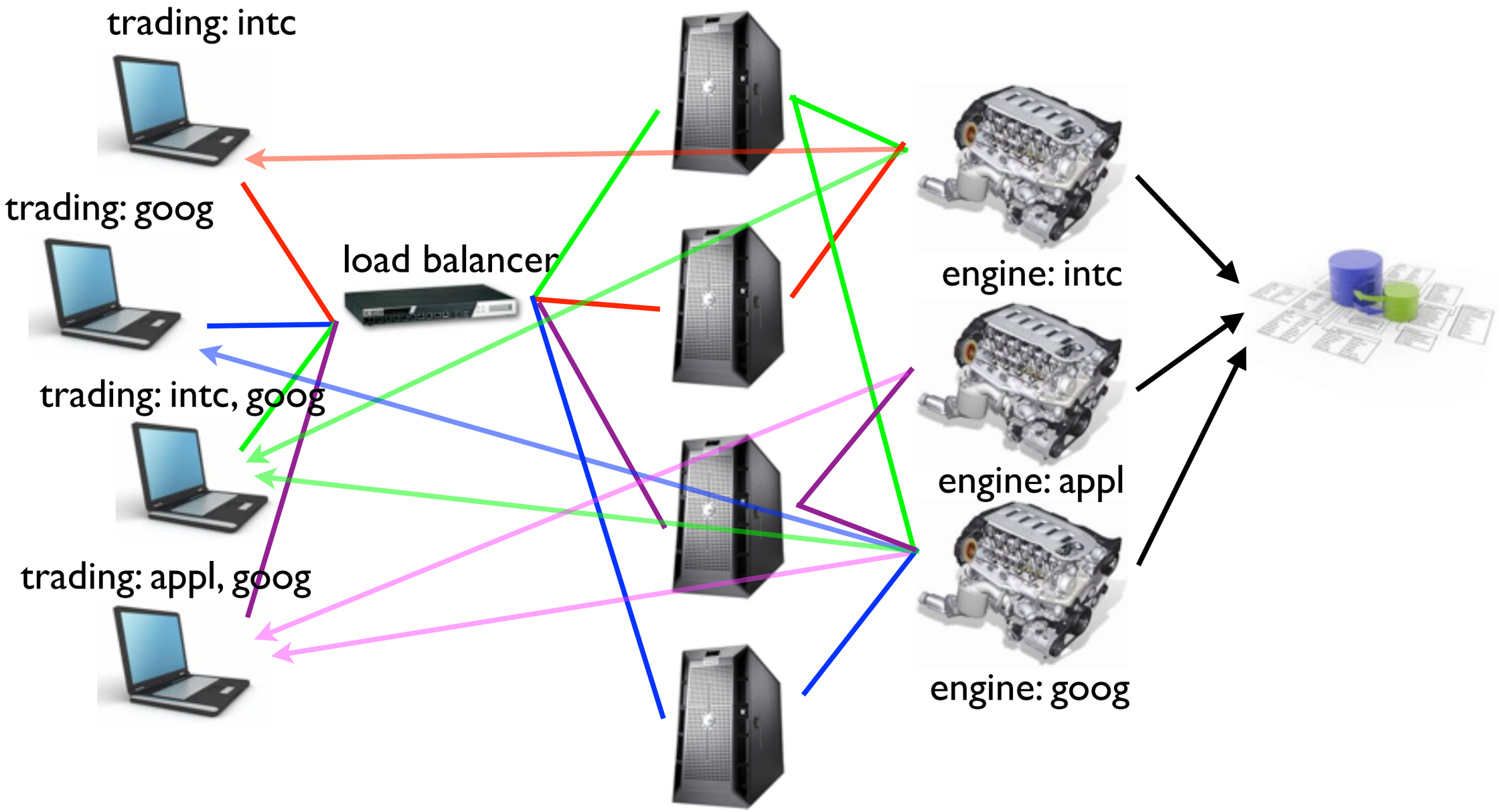
Design



Design


... logging
processing...
asynchronous delivery of
quotes, state change, etc.

Scaling


trading: intc
trading: goog
load balancer
trading: intc, goog
trading: appl, goog
engine: intc
engine: appl
engine: goog

Engine Design

- Last match price
- Last order ID
- List of Market Buy Orders
- List of Market Sell Orders
- List of Limit Buy Orders
- List of Limit Sell Orders
- List of Stop Buy Orders
- List of Stop Sell Orders

Engine Design

- When a order arrives it is added in the appropriate list in the proper order (by sequential order id)
- Engine tries to find a match between limit orders, between limit and market, and between two market orders. Limit buy orders with highest price are given precendence. Limit sell order with lowest price are given precedence, else order is FIFO. Orders can be matched partially. Leftover orders stay in their list.
- Last match determines new trading price for market orders
- For evey match, the engine checks for stop orders that meet condition, and they are moved to the appropriate lists
- Engine logs the order and the changes it caused
- Engine notifies all clients of the changes