Risk Management

And

Trading Systems

April 1, 2020

COLUMBIA UNIVERSITY
IN THE CITY OF NEW YORK

Overview

In this lecture we will explore concepts of operational risk management, explore the types of controls to mitigate these risks, and discuss strategies for placement in modern trading system software designs.

Risk Management: Why Do We Care?

- Humans (Traders, Quants, Programmers) make mistakes
- Increasing system complexity → emergent behaviors
- Public perception (the WSJ test)
- Regulatory Issues
  - ✓ Inquiries
  - ✓ New regulations
  - ✓ Fines
  - ✓ Real and opportunity costs
  - ✓ Operational Risk Capital Requirements
- When "Everything is broken all the time", managing operational risk must become a core competency

Regulation

- SEC, FINRA, Federal Reserve, Prudential Regulatory Authority (UK), FCA (UK), BaFin (Germany), SFC (Hong Kong) all considering expanded requirements to promote stable markets
- RegSCI: prescribed minimum disaster recovery requirements
- CFTC RegulationAT
- German HFT rules: algo tagging, etc
- BaFin Systems & Controls Guidelines – extensive required environmental, process and run time trading controls
- Developer registration
- And more…

We can solve our own problems,

or others will (attempt to) solve them for us

Errors and Costs

Loss of connectivity to data center	$2mm
Inadvertent replay of orders	$10mm
Reg NMS violations and inconsistent information to clients	$14mm
Alleged misrepresentations to clients about order routing	$154mm
Incorrect release processes and uncontrolled trading systems	$400mm

Types of Risks, Types of Controls

Risks	Controls
Human error	Value Checks/Limit Controls
Environmental	Processes and Procedures
Informational	Policies and Processes
Fraud	Logic constraints, Surveillance
Algorithmic / Logic	Run-time behavior controls and limits

Human Error - Limit Controls (aka “Fat Finger”)

- Max Order Size
- Daily Client Limits
- Daily Trader Limits

- Maximums expressed in different dimensions
  - ✓ Quantity (shares, contracts, etc.)
  - ✓ Value($)
  - ✓ Risk measure (dv01, delta, etc.)

Environmental

- System Access
- System monitoring
- Software release processes
- Environmental leakage
- Resiliency
- Documentation / training

Informational

- Marketing material versus functionality: does the label match what's in the tin?
- Information dissemination : do all clients get the same message?
- Asymmetry of offered features : Do some clients get special functionality (e.g. different order types etc)?
- Best execution policy

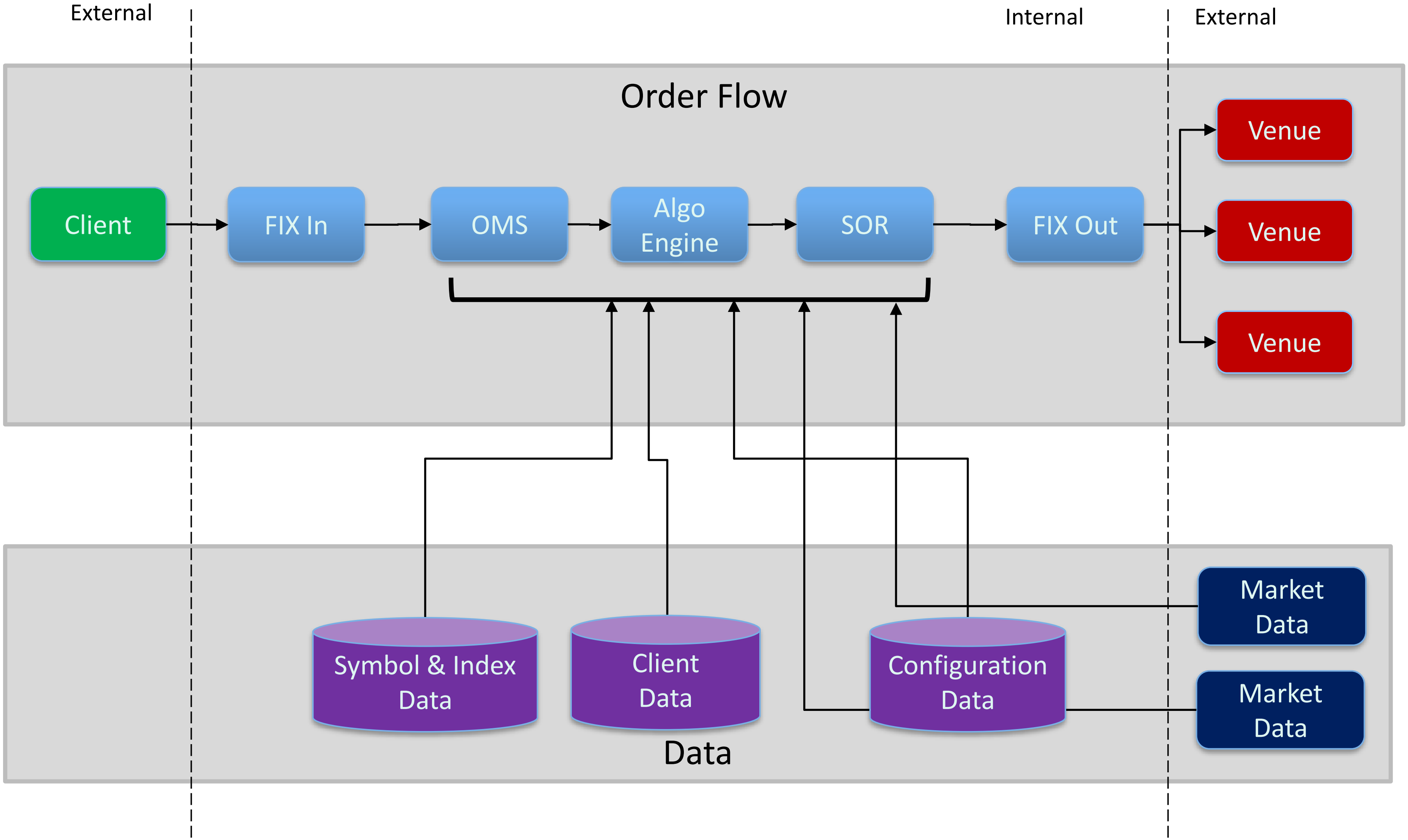
Fraud

- Manipulation – placing orders with no direct economic benefit to distort or mislead the market
- How to detect in fragmented markets?
- Surveillance

Example Runtime Controls

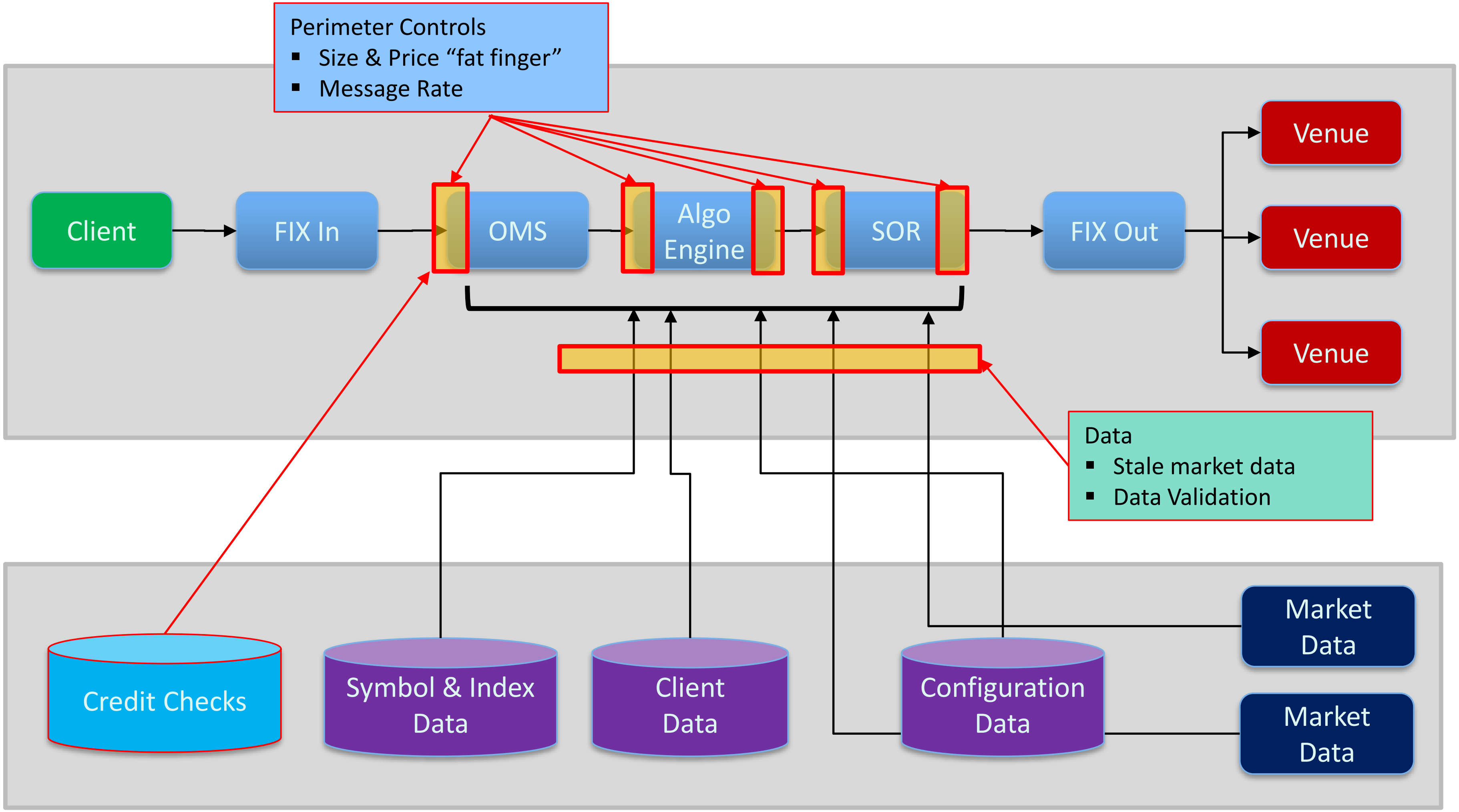
Price movement	Has the price moved more than some % within a certain time period relative to historical norms?
Price versus benchmark	Has the price moved away from a specific benchmark (e.g. price at order arrival time)?
Size, value, risk checks	Similar to fat finger checks: are the order(s) we are sending too large by one or more measures?
Message rate	Are we sending too many orders?
Valid market data	Do we really know what the current market price is? What if we don’t?
Stale order checks	Are the orders I'm sending or receiving older than is reasonable?
Heartbeats	Are all of our system components running? What if they aren’t?

Order Execution – Flows



Columbia University
In the City of New York

Order Execution – Runtime Controls



Perimeter Controls

- Size & Price "fat finger"
- Message Rate

Data

- Stale market data
- Data Validation

Scenarios

Scenario 1: Communication

Orders sent, but fills don't come back to the system

1. An algorithm wants to buy 1000 shares of a stock
2. Sends an order: buy 1000 @ market
3. The order is executed, but the system does not receive the fill (confirmation of execution)
4. So it tries to resend, go to step 2

Potential Effects

- Strategy buys significantly more stock than it needs to
- Potential material losses
- Impact to the stock (market orders will eventually force the price higher)
- Regulatory scrutiny

Scenario 2: Order Replay

Procedures to ensure a clean start of day are not followed during a system upgrade.

1. System starts up
2. It checks to see if there are any orders in the system that need to be processed (standard behavior, in case of a crash, etc.)
3. It finds orders in the system and resends them to the market
4. Many of the orders execute
5. Because other parts of the system did not know the orders, the firm is not aware of the executions until next day

Potential Effects:

- Potential material losses
- Impact to market as extra orders get sent
- Regulatory inquiries

Scenario 3: Bad Market Data

Statistical arbitrage strategy

+ bad market data

= prices that are too good to be true

1. Market is 100 – 100.01
2. Market data feed says 10 – 10.01
3. 10.01 looks great! Buy!
4. System buys stock all the way to risk limit. But at 100.01, 100.10, 101, 105, 110…
5. When buying stops, price reverts

Potential Effects:

- Significant market move
- Theoretical profit of ~$90/sh → real loss

Scenario 4: Environmental Failures

During a software release, not all servers are updated with the new version of the software.

Potential run time effects depend on nature of change:

- New release just doesn’t work at all
- New release doesn’t work (but only for some orders… potentially difficult to troubleshoot)
- New release interacts with old code in an unexpected way, resulting in orders being sent to market erroneously.

Potential Effects: Hard-to-troubleshoot errors, severe losses, market impact, regulatory inquiry

Scenario 5: Feedback Loops…

A trader splits a large order into smaller orders and sends them in parallel, effectively competing with each other.

1. Time 0: 10,000 shares print

2. Time 1: 5 algos at 20% volume each trade 2,000 shares

   Total volume = 10,000 + 5 * 2,000 = 20,000 shares

3. Time 2: 5 algos at 20% volume each trade 4,000 shares

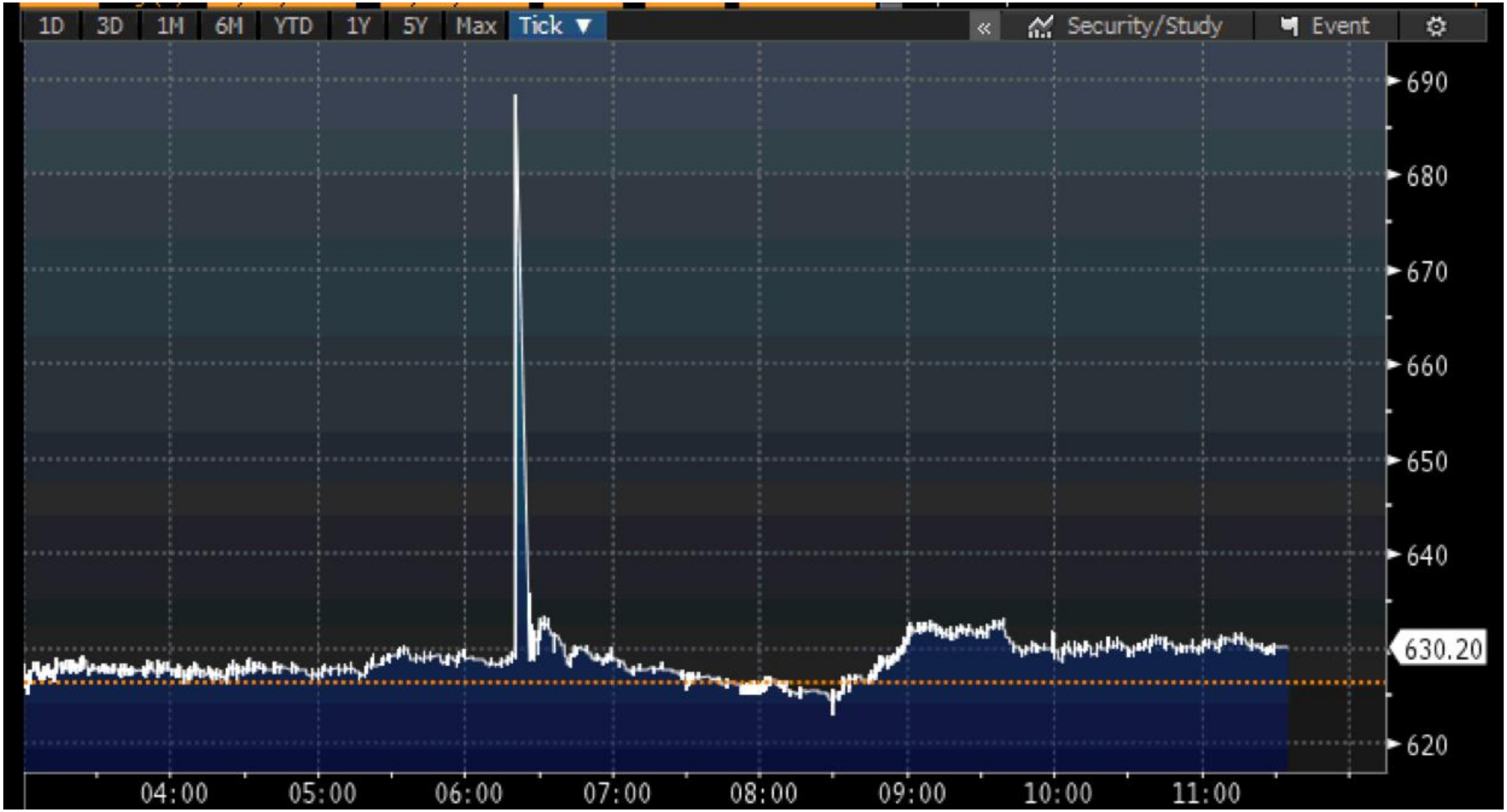
   Total volume = 20,000 + 5 * 4,000 = 40,000 shares

4. Time 3: 5 algos each trade 8000 shares

   Total volume = 40,000 + 5 * 8,000 = 80,000 shares

That's just the first 500-1000 milliseconds…

…Make Bad Things Happen



COLUMBIA UNIVERSITY
IN THE CITY OF NEW YORK

In Summary: Confidence and Control

- Humans (Traders, Quants, Programmers) make mistakes
- Increasing system complexity → emergent behaviors
- Public perception (the WSJ test)
- Regulatory scrutiny

Good operational risk management can

a) be the difference between profitability and unprofitability

b) help to maintain (restore?) trust in stable, functional markets

Questions?