CS 349F: Technologies for Financial Systems

Instructors: Balaji Prabhaparker and Mendel Rosenblum
Course Logistics

Meeting time/place: Mondays, 4.30—5.50pm; 200-002
Course Admin: Kara Marquez, Packard 267
Course website: http://web.stanford.edu/class/cs349f/

- Will mostly have a few papers and other reading materials
- Class mailing list: cs349f-spr1819-staff@lists.stanford.edu

Off hours:
Prabhakar: Mon, 3—4pm, Packard 269
Rosenblum: Wed, 1—2pm, Gates 3A-354

Grading basis: S/NS (Pass/Fail); to get an S, you need
- Attendance in 8 out of 9 lectures (sign up sheet)
- A 2-page report by 5pm, June 7th, 2019 (details will be emailed in advance)

Event of interest: Apr 11th, 2019, Symposium on Faster, Fairer Fintech
A Brief History of Financial Trading

People have been trading forever; however, some noteworthy milestones...

• 1100s—1200s: France, courretiers de change, managed agricultural debt (by trading it)
• 1300s: “Merchants of Venice”, Florence, Genoa, etc: Trading government securities (bonds)
• 1400s—1500s: First “stock” markets: Belgium (Antwerp), Rotterdam, Flanders, Netherlands
• Antwerp had the wealthy Van der Beurze family; hence early stock markets called Beurzen (not the root of the word Bourse!)

The world’s first publicly traded company (more or less): East India Company

• Mutualize risk
  – Putting all money into one ship → bad idea
  – Buy shares in multiple companies and their ships → better idea
• Dutch East India company officially became the first publicly traded company by selling shares on the Amsterdam Stock Exchange in 1602

Today...
A Brief History of Financial Trading

Major stock exchanges in the world (ranked by market cap)

- NYSE ($30.9B), NASDAQ ($10.8B), Japan ($5.5B), Shanghai ($4B), HKEX ($3.9B), EuroNext ($3.9B), London ($3.7B), Shenzhen ($2.5B), Toronto ($2B), Bombay ($2B), Indian NSE($2B), Deutsche Borse ($1.86B), ...

Chicago Mercantile Exchange ($59.6B)

- Commodities, Derivatives
- The “biggest financial exchange you’ve never heard of.” (Economist)

Foreign Exchange

- The biggest asset class, the most liquid
- Operates 24 hrs except on weekends
  → $5.09T per day in Apr 2016 (see Wiki page on ForEx)
To a first order approximation, financial trading can be broken into two main computational loops

1. Fast loop: At trading venues like NYSE, NASDAQ, CME, etc
2. Slow loop: When a market participant runs large-scale computations on market data to devise their trading strategies or algorithms

Key concept that comes up in modern financial trading is high frequency trading (HFT)

- Some people think this is bad since HF is not required for discovering prices or for providing liquidity
- Others (including some amongst the regulators) think it does make the market more efficient
- Related concept: arbitrage

Let’s look at Financial Trading from a CS point of view...
Financial Trading Consists of Two Computation Loops

Fast Loop: Orders ↔ Market Data

- **At exchanges and trading venues:** Market participants respond to market data, place orders to execute trades in real-time
- **Low latency is critical:**
  - FPGAs perform trades at great speed, bring determinism
  - network is carefully engineered to *eliminate jitter*
  - excess bandwidth and compute power ensure low latency
Jitter is a Key Problem for Financial Exchanges
Timestamp transactions at gateways to establish precise order of arrival

Resequence transactions in a "reordering buffer" before execution

Timestamp order books and ...

... release them *simultaneously* at (geographically) different locations ➔ example of a "smart contract"
SAN FRANCISCO — Computer scientists at Stanford University and Google have created technology that can track time down to 100 billionths of a second. It could be just what Wall Street is looking for.

System engineers at Nasdaq, the New York-based stock exchange, recently began testing an algorithm and software that they hope can synchronize a giant network of computers with that nanosecond precision. They say they have built a prototype, and are in the process of deploying a bigger version.

For an exchange like Nasdaq, such refinement is essential to accurately order the millions of stock trades that are placed on their computer systems every second.

Ultimately, this is about money. With stock trading now dominated by computers that make buying and selling decisions and execute them with blazing speed, keeping that order also means protecting profits. So-called high-frequency trading firms place trades in a fraction of a second, sometimes in a bet that they can move faster than bigger competitors.

The pressure to manage these high-speed trades grows when the stock market becomes more volatile, as it has been in recent months, in part to prevent the fastest traders from taking unfair advantage of slower firms. High-frequency traders typically ac-

Continued on Page A17
Financial Trading Consists of Two Computation Loops

Slow Loop or Training Mode: Market Data ↔ Trading Strategies and Algorithms

- **In large-scale compute clusters:** Process market data to devise proprietary trading strategies and fine-tune trading algorithms
- **Accurate market data capture and playback:** Relies on high-precision timestamping of market feeds
- **Large-scale computation and ML/ AI tools needed:** Where Cloud computing has an edge, but oftentimes market data is in the exchange colocation facility, not in the Cloud
  - Cloud Computing is built for “batch processing”, not for “real-time computation”
Bifurcations 50 Years Ago

Computing
- Centralized → Supercomputing/high-performance, real-time
- Distributed → Cluster computing/batch processing, high throughput
  Key constraint: Need to coordinate cluster, no time synchronization

Networking
- Circuit-switched → Real-time, high QoS guarantees
- Packet switched → "Best-effort service," random packet delays
  Key problem: Couldn’t get “jitter-free networks”
Self-Programming Networks

A quest to make networks

*autonomous*: network should sense and monitor itself; program and control itself

*interactive*: network should be simple and fun to use, especially for 3rd party users

SPN = Engineering + Design

**Autonomy**
- Systems
- Programs
- ML/AI Algorithms

**Interactivity**
- Visual
- Transparent
- “Friendly and chatty”
Self-Driving Cars

Plain Old Volkswagen Tuareg

Sensing + Control

Stanley
The Self-Driving Car
**New Functionality**
- Timestamping As A Service
- Fine-grained Network Telemetry
- App-Network Perf Monitoring

**Self-Programming Network (SPN)**

![Diagram showing the SPN architecture with components like Workload, Sense, Control, and New Functionality](image-url)
Self-Programming Network (SPN)

Workload

Plain Old Data Center

New Functionality
- Timestamping As A Service
- Fine-grained Network Telemetry
- App-Network Perf Monitoring

Sense, Infer, Learn and Control (SILC)

Sense

Control

Interactive Dashboard & Query Engine

Intuitive DB and QE
- Simple + visual + chatty
- App+network perf views

NIC-centric Architecture
- Sensing and control at NICs
- Smart NICs: big industry trend

Data and ML Intensive
- Use data and NNs to accelerate learning and for real-time processing
**NIC-centric Architecture:** Network Telemetry From The Edge

- **From total time in the network, determine time spent in each switch**

**Advantages of NCA**

- **Low latency:** Use NIC hardware timestamps and avoid software stack latency
- **Scalability:** Edge observations are sufficient for whole network telemetry
- **Smart NICs:** Big industry trend + can run powerful ML algorithms in place
NIC-centric Architecture: At the Computing-Networking Boundary
Rest of the Lectures

• Combination of presentations by me, Mendel and industry experts
• Networking and Cloud Computing basics
  • Switches, switch architectures, data center interconnect fabrics
  • Cloud computing: virtualization and distributed computing
  • Building jitter-free networks and scalable distributed systems
    – The role of time and synchronization
    – NIC-based telemetry
    – Smart NICs
    – Scaling consensus
• Inference, ML and AI algorithms
  • In clock sync, network telemetry
  • Guest lecture: Use of AI in devising trading algorithms
• How exchanges and trading work
• Distributed ledger technology
  • How it works
  • Its use in ForEx trading