Chapter 1

David Leinweber

President, Leinweber & Co.

I wish I could tell one of those stories about how, when I was in the eighth grade, I noticed a pricing anomaly between the out-of-the-money calls on soybean futures across the Peruvian and London markets and started a hedge fund in my treehouse and now own Cleveland. But I can't. In the eighth grade I was just a nerdy kid trying to keep my boisterous pals from blowing up my room by mixing all the chemicals together and throwing in a match. In fact, I really can't tell any true stories about eighth graders starting hedge funds in treehouses buying Cleveland. Make it sophomores in dorm rooms who buy chunks of Chicago, Bermuda, or the Cayman Islands, and we have lots of material.

A Series of Accidents

My eventual quantdom was not the culmination of a single-minded, eye-on-the-prize march to fulfill my destiny. It was the result of a series of accidents. In college, my interest in finance was approximately zero. I came to MIT in 1970 as a math major, as did many others, because I didn't know much about other subjects like physics or computer science. I quickly discovered the best gadgets were outside the math department. And the guys in the math department were a little weird, even by MIT standards. This was back when even a pretty crummy computer cost more than an average house. A good one cost millions, and filled a room the size of a basketball court. MIT, the ultimate toy store for geeks, had acquired a substantial inventory of computing machinery, starting as soon as it was invented, or sooner, by inventing it themselves. The professors kept the latest and greatest for themselves and their graduate student lackeys, but they were happy to turf last year's model to the undergrads.

Foremost among these slightly obsolete treasures was the PDP-1-X, which is now justly enshrined in the Computer Museum. The PDP-1-X was a tricked-out version of the PDP-1, the first product of the Digital Equipment Corporation (DEC). The story of DEC is an early computer industry legend now fading in an era where many people believe Bill Gates invented binary numbers.

DEC founder Ken Olsen worked at MIT's Lincoln Laboratory, where the Air Force was spending furiously to address a central question facing the nation after World War II: "What do we do about the Bomb?" Think about the air war in World War I. There were guys in open cockpits wearing scarves yelling, "Curse you, Red Baron!" By the end of World War II, just 30 years later, they were potential destroyers of worlds. Avoiding the realization of that potential became a central goal of the United States.

If a Soviet bomb was headed our way, it would come from the north. A parabolic ballistic trajectory over the pole was how the rockets of the era could reach us. This begat the Distant Early Warning (DEW) and Ballistic Missile Early Warning (BMEW) lines of radars across the northern regions of Alaska and Canada. The DEW and BMEW lines, conceived for military purposes, drove much of the innovation that we

see everywhere today. Lines of radars produce noisy analog signals that need to be combined and monitored.

Digital/analog converters were first on the DEW line, now in your iPod. Modems, to send the signals from one radar computer to others, were first developed to keep the Cold War cold. Computers themselves, excruciatingly large and unreliable when constructed from tubes, became transistorized, and less excruciating. This is where Ken Olsen comes in. Working at MIT to develop the first transistorized computers for the DEW line, he and his colleagues built a series of experimental machines, the TX-0 (transistor experiment zero), the TX-1, and the TX-2. The last, the TX-2, actually worked well enough to become a mother lode of innovation. The first modem was attached to it, as was the first graphic display, and the first computer audio.

Olsen, a bright and entrepreneurial sort, realized that he knew more about building transistorized computers than anyone else, and he knew where to sell them—to the U.S. government. Federal procurement regulations in the early 1960s required Cabinet-level approval for the purchase of a computer, but a Programmable Data Processor (PDP) could be purchased by garden-variety civil servants. Thus was born the PDP-1 and its successors, up to the PDP-10, like the one at Harvard's Aiken Comp Lab used by a sophomore named Gates to write the first Microsoft product in 1973.

Today, almost any teenage nerd has more computational gear than they know what to do with. But in the 1970s, access to a machine like the PDP-1, with graphics, sound, plotting, and a supportive hacker[1] culture was a rare opportunity. It was also the first of the series of accidents that eventually led me into quantitative finance.

I wish could I could say that I realized the PDP-1 would allow me to use the insights of Fisher Black, Myron Scholes, and Robert Merton to become a god of the options market and buy Chicago, but those were the guys at O'Connor and Chicago Research and Trading, not me.

I used the machine to simulate nuclear physics experiments for the lab that adopted me as a sophomore. They flew down to use the particle accelerators at Brookhaven National Lab to find out the meaning of life, the universe, and everything else by smashing one atomic nucleus into another. Sort of a demolition derby with protons. But sometimes

a spurious side reaction splatted right on top of whatever it was they wanted to see on the glass photographic plates used to collect the results. My simulations on the PDP-1 let us move the knobs controlling electromagnets the size of dump trucks so the spurious garbage showed up where it wouldn't bother us. It was fun to go down to Brookhaven and run the experiments.

The head of the lab was a friendly, distinguished Norwegian professor named Harald Enge. As a young man, Harald built the radios used by the Norwegian underground group that sank the ship transporting heavy water to Hitler's nuclear bomb lab. Arguably, this set the Nazi A-bomb project back far enough for the Allies to win the war, so we were all fans of Harald. He drove a Lincoln so large that there were many streets in Boston he could not enter, and many turns he could not make. It was worth it for safety, he explained. As a nuclear scientist who spent his career smashing one (admittedly very small) object into another, he explained that he had an innate sense of the conservation of momentum and energy, and was willing to take the long way around to be the big dog of p and E.

Senior year, I planned on sticking around for graduate school as a physics computer nerd, a decision based more on inertia than anything else. Then I met the saddest grad student at MIT. The nuclear physicists were replacing those glass photographic plates with electronic detectors. These were arrays of very fine wires, arranged very close to each other to emulate the fine resolution of photography. This grad student had made a 1,024-wire detector, soldering 1,024 tiny wires parallel to each other, then 2,048 wires. He was currently toiling over a 4,096-wire version. The work was so microscopic that a sneeze or quiver could screw the whole deal. He'd been at it for a year and half.

At around the same time, Harald showed me, and the other undergrads considering physics graduate school, a survey from the American Institute of Physics of the top employers of physics PhDs. An A in the survey meant, "Send us more," while a D meant, "We're trying to get rid of the ones we've got." There were hundreds of organizations. There were no "As." This two-part accident, meeting the grad student in 4K wire hell and seeing that I would be lucky to find a job in a place like Oak Ridge (which, to the eyes of a New York City kid, looked like the moon but with trees), sent me to computer science graduate school, a step closer to becoming a quant.

Harvard University, the school up the road that once wanted to merge with MIT and call the combination *Harvard*, had a fine-looking graduate program in computer science with courses in computer graphics taught by luminaries David Evans and Ivan Sutherland. Harvard not only let me in—it paid for everything. Instead of making a right out my front door, I'd make a left. I could stay in town and continue to chase the same crowd of Wellesley girls I'd been chasing for the previous four years.

I showed up in September 1974 and registered for the first of the graphics courses. Much to my surprise, my registration came back saying the courses weren't offered. I had discovered the notorious Harvard bracket. The course catalog was an impressive, brick-sized paperback with courses covering, more or less, the sum of human knowledge. Many were discreetly listed in brackets. The brackets, I discovered, meant, "We used to teach this, or would like to. But the faculty involved have died or otherwise departed. But it sure is a fine-looking course." The Harvard marching band used to do a salute to the catalog, where about half of the band would form brackets around the rest, and the people inside the brackets would wander off to the sidelines, leaving nothing.

My *de facto* advisor, Harry Lewis, then a first-year professor, later Dean of the College, suggested that the accident of the missing graphics track allowed me to sample the grand buffet of courses actually taught at the university. The Business School had a reputation for good teaching, and offered courses with enough math to pass my department's sniff test. So off I went across the river for courses in the mathematics of stock market prices and options. They were more of a diversion than an avocation, but the accident of the brackets had more influence subsequently than I could have imagined at the time.

Harry also enlisted me as the department's representative on the Committee on Graduate Education, which gave me a reason to hang out in the dean's office. He was on the board of the RAND Corporation in Santa Monica, and suggested it might be a nice place to work, right on the beach with no blizzards. I put it on my list.

Grey Silver Shadow

When the time came to find a real job, I was going out to UCLA to interview for a faculty position, and I added RAND to the schedule. UCLA

told me to stay in the Holiday Inn on Wilshire Boulevard, rent a car, and come out in February of 1977. On the appointed day, I opened my door in Inman Square to drive to Logan Airport and saw that a ferocious storm had buried all the cars up their antennae. I dragged my bag to the MTA station, and dragged myself onto a delayed flight to Los Angeles.

At this point, I had never been west of Pennsylvania Dutch country. Leaving the tundra of Boston for balmy Los Angeles was an eye-opener from the beginning. At LAX, I went to retrieve the nasty econo-box rental car that had been arranged for me. I was told they were fresh out of nasty econo-boxes, and would have to substitute a souped-up TransAm instead. Not that I knew what that was. It turned out to be a sleek new metallic green muscle car, with a vibrating air scoop poking up through the hood. I was a nerd arriving in style. Leaving the airport, I found myself on the best road I'd ever seen, the San Diego Freeway, I-405. This was in the pre–Big Dig days of Storrow Drive, so my standard for comparison was abysmally low. The 405 made a transition via a spectacular cloverleaf onto an even better road, the Santa Monica Freeway. I later learned that this intersection, designed by a woman, is considered an exemplar of freeway style. It sure impressed me.

The UCLA recruiter's hotel advice was flawed. There were *two* Holiday Inns on Wilshire Boulevard. One near campus, the other further east, across the street from the Beverly Wilshire Hotel near Rodeo Drive, the hotel later made famous in *Pretty Woman*. I drove through Beverly Hills in blissful ignorance, thinking it was a pretty fancy neighborhood for a college. Street signs in Boston were mostly missing. Here, they were huge, and placed blocks ahead, so drivers could smoothly choose their lane. The sidewalks actually sparkled. Beverly Hills uses a special high-mica-flake-content concrete to do this. There were no sixties acid burnouts jaywalking across my path. Cars were clean, new, fancy, and without body damage. We weren't in Cambridge anymore.

I steered my rumbling TransAm into the parking lot for the hotel, and got out. I wore the standard-issue long-haired grad-student garb of Levis, flannel shirt, and cheap boots. A white Lamborghini pulled in, just in front of me. This was the model with gull wing doors, selling for about half a million, even then. I'd never seen anything like it outside of a Bond movie. The wings swung up, and two spectacularly stunning starlet types, in low-cut tight white leather jumpsuits, emerged. Big hair,

spike heels, lots of makeup. In Cambridge, it was considered politically incorrect for women to look different from men while wearing clothes. In LA this did not pose a problem.

Before I could resume normal respiration, a well-dressed gent walked up and dropped a set of keys in my hand. "Grey Silver Shadow," he said. I had no idea he was talking about a car so lavishly priced that I could not buy it with three years' salary for the UCLA and RAND jobs combined. A quicker thinker would have said "Yes, sir!" and driven the Rolls off to Mexico with the Lamborghini girls. I meekly explained that I wasn't the attendant and gave the keys back. This remains one of my great regrets.

So how does this advance the plot of how I became a quant? I ended up at RAND doing nice civilian work such as artificial-intelligence-inspired analysis of econometric models for the Department of Energy and the EPA and helping with the design of a storm surge barrier for the Dutch water ministry. All very interesting, but fairly remote from quantitative finance. In 1980, Reagan won the election, and promised to abolish both the EPA and the DoE. He didn't quite do that, but the cash flow to RAND from those agencies slowed to a trickle. The Dutch stopped analyzing and started building the Oosterschelde Storm Surge Barrier.[2] I was drafted into the military side of RAND. There were classified and unclassified sides of the building, separated by thick secure glass doors operated by guards. I moved over, and filled out the paperwork to upgrade my security clearance to Top Secret. Everyone needed a secret just to get in the building.

The project I was handed[3] could have been called "We're kind of worried about the space shuttle." In 1980, the shuttle was two years late, $5 billion over budget, and 40,000 pounds overweight. The Air Force and the Defense Advanced Research Projects Agency, which were the biggest customers, were justly concerned. As things turned out, they were right. According to the schedule that accompanied the sales pitch, the shuttle was to have flown 400 flights in its first ten years. The most recent launch on December 9, 2006, after almost 26 years, was number 116. The fleet was grounded for two year-long periods after the accidents in 1986 and 2003. All of this was not unanticipated by the engineers in 1980.

The pacing-size payloads for the shuttle—the ones it was too heavy to carry—were experimental platforms for testing sensors designed to

be operated by people, the mission specialists. They would interpret the results of experiments and decide on the next steps. Now, it looked like they wouldn't be there. Ground links weren't an option. This left the Pentagon with a problem. Here is a complex system, the sensor platform, getting instructions over wires, and sending back results that require analysis and decision in real time. Lucky for me, that also turned out to be a description of financial markets and trading rooms. When the people can't be there, the technological solution is some sort of real-time artificial intelligence (AI). The state of the art of AI at the time ran toward theorem proving and dealing with other static problems. My mission was to find promising places to foster the growth of real-time AI and have the boys in the five-sided nuthouse write checks to make it happen.

In the course of that work, I visited all of the AI companies that were too big to fit in a garage. Most were scattered in the vicinity of MIT, Stanford, and CMU. They had cryptic sci-fi names like Intellicorp, Inference, Symbolics, and LISP Machines.[4] When you show up with the Pentagon's checkbook, you get the good lunch. In this case, that meant "not from the vending machine." So I spent quality time with the top AI nerds and their business chaperones on both coasts. Sometimes there were promising technologies, but there was always interesting company. This was the same crowd that had formed around the PDP-1 at MIT—always in spirit, and often in person. I felt right at home.

Destroy before Reading

This went on for a couple of years, working on the rocketry aspects of the "What about the shuttle" project when I wasn't sharing take-out Chinese food with the AI guys. We wrote up what we found. Most of it was lightly classified by the Air Force officers at RAND. Lightly classified means *secret* or *confidential*. The latter is rarely used. Rumor had it that the Soviet ambassador was cleared for confidential. Dealing with secret material was not all that onerous. You could carry it on commercial aircraft, inside double envelopes and with a permission slip. You could read it in a RAND office with the window open.

Top Secret, and beyond, is another world entirely. It's not quite "Destroy before Reading," but it's close. No civilian planes to move it around. Military escort required. Go down to a vault to read it. Don't write anything down. Expect your phone to make funny noises and your mail to be late. I was glad not to have to deal with it. But in 1983, President Reagan gave his "Star Wars" speech, and everything having anything to do with the military use of space became so highly classified it made your teeth hurt.

I had a file cabinet in my office, with a large collection of articles from *Aviation Week* and the *New York Times*. Nothing classified at all. I kept that in my "secret locker" down the hall. My lunch was in the cabinet's bottom drawer, along with beverages and salty snacks for the after-hours time on the beach. One day, two guys in blue uniforms came in from the USAF Space Division in El Segundo. They loaded my file cabinet onto a cart and the following conversation ensued.

Me:	"Hey, there's nothing much in there except for stuff from *Aviation Week*."
Blue Suiter:	"They publish a lot that they shouldn't publish."
Me:	"Maybe so, but the cat's out of the bag once they print it. Do you know how many copies of *Aviation Week* go to the Soviet Embassy?"
Blue Suiter:	"Nope"
Me:	"I do, 285. 'Think there's anything in there they don't already know?"
Blue Suiter:	"We've got our orders."
Me:	"Okay, but can I keep my lunch? Want some snacks?"

If that wasn't weird enough, a few weeks later I was called in to the classification office to review a paper I'd written for an academic conference on space and national security. After the file cabinet experience, I had taken extreme care to use only the most publicly available material I could find, and to avoid *Aviation Week* entirely. For security reasons, we'll give RAND's Air Force classification officer a secret identity: Major Pain.

Major Pain:	"I have some problems with your paper."
Me:	"For instance . . ."

Major Pain: "Over here, where you talk about the 'National Technical Means of Verification.'" (1980s diplomat-speak for spy and warning satellites)
Me: "That's straight from a speech Jimmy Carter gave on television. That's why it's in quotation marks next to his name."
Major Pain: "I know. He said a lot he shouldn't have said."
Me: "With due respect, he was commander-in-chief and you're a major."
Major Pain: "But I'm *your* major, and this conference is next week."
Me: "You win, Jimmy's gone. Anything else?"
Major: "Of course."

It was time to become a civilian. I called my pals at the AI companies and made a beeline for the door. I ended up working for Steven Wyle,[5] the chairman at LISP Machines Inc., who conveniently had set up offices right in Los Angeles. (Most of the company was back in Cambridge.) LISP Machines had some of the most promising real-time AI capabilities, which ran on the special-purpose LISP computer that LMI and its rival Symbolics both manufactured. That there were two companies that licensed the same technology from MIT at the same time was a testimonial to the inability of nerds to get along.

A Little Artificial Intelligence Goes a Long Way

LMI was founded by Rick Greenblatt, the machine's inventor. He had a habit of leaving Nutty Buddies (vending-machine ice cream cones topped with chocolate and nuts) in his front pocket and forgetting about them. This made for a distinctive fashion statement. He was also an early avatar of the free software, open-source movement, which later became GNU and Linux. Richard Stallman was encamped there. Symbolics, founded by the AI Lab administrator, who wore a suit with no food on it, was more businesslike.

Both companies quickly fell victim to the fate of computer firms that make special-purpose machines. If you ever want to start one of

these, do something with better prospects of success like invading Russia in winter.

AI was getting great press in the 1980s, better than it deserved. Business magazines hawked the "Breakthrough of the Century" and "Machines That Think." In fact, AI's successes and capabilities were more modest, but it was good at making computers easier to use. All the noise attracted people from places other than the computer research labs that formed the original market for LISP machines (and Symbolics, and the rest). At LISP Machines, my portfolio included space applications, communications, and all the sorts of applications people at RAND worried about.

When people from Wall Street started showing up, the boss asked, "Who can talk to these guys?" and I finally got to make some use of my off-major experience in graduate school. Options guys from Chicago? I knew delta wasn't just an airline. Traders from Wall Street? I knew a bid from an ask, and an option from a future. By default, I became the in-house ambassador to finance.

As the hardware firms were thinning out, I went across the street to Inference Corporation, a software-only AI firm that shared investors (and at one point, offices) with LISP Machines. Another fortunate accident was that they had just hired Don Putnam as president, luring him away from the institutional financial service firm, SEI Investments. When I met Don, he hired me on the spot and told me to forget about satellites and the DoD, and spend all my time on finance. No more Major Pains. It sounded good to me.

Inference's product was called the *Automated Reasoning Tool*, really a sort of syntax relief for LISP. It had modules for nearly every artificial intelligence technique. NASA was the biggest customer. Don worked some kind of deal with Quotron,[6] then the major market data vendor and conveniently located down the street, that allowed us to use actual market data to try out our wacky ideas. This might have been one of the first times anyone actually tied the consolidated feed to an expert system.

Our modest efforts at a prototype were immodestly called the ART Quotron Universal Investment Reasoning Engine—AQUIRE, which had a nice Gordon Gekko feel to it (even though the actual Gordon from *Wall Street* was a year away, in 1987). As it turned out, the "Universal Investment Reasoning" demonstrated in AQUIRE consisted of variations

on crossover rules—comparisons of moving averages. These seemed to be a favorite of the New York visitors, and were easy to program. Many of the traders had their own secret-sauce variations on this theme, combining different averaging intervals and lags. The former math professors from Chicago preferred complex arbitrage relations and formulae involving the entire Greek alphabet, which took more time to program.

All of this ran on playbacks of recorded data, so we could fix our mistakes and replicate the examples our customers showed us. It also pointed up the tragic flaw in LISP based trading systems: garbage collection. AI programs tended to grab and then abandon large chunks of memory.[7] The system would periodically take a snapshot of the memory used by currently active variables, and collect the "garbage" left unused and return it to the pool of available memory. This freed the programmers from the task of memory management, but had the unfortunate side effect of causing the machine to "take a moment" while it collected itself. These moments could extend into many minutes of waiting—the kiss of death for real-time trading applications in LISP.

Garbage collection was only one of the features of the general-purpose AI tools that rendered them less than desirable for financial applications. The baggage they carried that allowed solutions to everything from chess problems to theorem proving to network analysis was too much for a fast, focused effort on trading. Don and I tried to change this at Inference.

In 1987, after months of discussion with the chairman, we parted company. Don Putnam founded the company that became Putnam-Lovell. Its first investment was in Integrated Analytics, which Dale Prouty and I founded to deliver the specialized and less-filling expert system environment needed for financial applications. Years later we published a paper, "A Little Artificial Intelligence Goes a Long Way on Wall Street,"[8] on the details.

How Do You Keep the Rats from Eating the Wires?

Shortly after we started the company, a colleague from the AI group at Arthur D. Little, the venerable Cambridge consulting firm, asked me

to fill in for him at the last minute at a technology session at a finance conference being held in Los Angeles. His dog was sick. The topic was a generic "AI on Wall Street," the last one in a catch-all session. The other speakers were from brokerage firms, plus someone from the American Stock Exchange. The audience was about 75 technology managers.

I'd planned sort of an AI 101 talk, going over various solution methods, forward and backward chaining, generate and test, predicate logic, and the rest. While I was reviewing my slides, the Amex guy was showing photos of how they'd managed to install cables in a building designed in the nineteenth century. Then he took questions.

"How do you keep the rats from eating the wires?" A great question. The answer is that there are certain plastics that rats don't seem to like, and that's the wire to use. I realized the whole thing with the back-chains and the predicate logic wasn't going to play here. Instead, I followed the lecture formula espoused by some of the best, wrapping the content in jokes. The content boiled down to "computers are pretty good at manipulating other computers—you have better things to do." The jokes were sufficiently amusing that I didn't come off as a complete conehead. Someone from Cantor Fitzgerald, then based in Los Angeles, even invited me over to do it again for their trading room.

Cantor Fitzgerald occupied several floors at the top of a prime building in Century City, adjoining Beverly Hills. Bernie Cantor's collection of Rodin statuary filled a large portion of the main floor. I'm not talking about little table-top items. Rodin often worked larger than life unless you live with an NBA team. We were suitably impressed. Our host, Phil Ginsburg, a former professor from Northwestern, had been hired as the chief in-house nerd. There were white-jacketed waiters delivering beverages and snacks, including frozen grapes. (The frozen grapes are a pretty good idea. Use seedless, and let them thaw a bit.) There was no Mr. Fitzgerald at Cantor Fitzgerald. Bernie Cantor thought that just plain *Cantor* was too ethnic sounding. People called him Mr. Cantor, never Bernie, when he was around, and Bernie at all other times.

We showed our MarketMind prototype to the equity traders, who were thrilled. In 1988, market data systems were just beginning to show charts. They were limited to one stock at a time, and one type of chart at a time. MarketMind let them watch hundreds of stocks, with as many types of chart as the machine could handle. The program figured

out which of the many thousands of chart/symbol combinations were interesting. The machine in this case was a Sun Unix workstation, PCs running the then-current DOS 4.0 being hopelessly inadequate. The charts they wanted included all flavors of intraday technical analysis, mostly variations on crossover rules, with many filigrees—nothing we couldn't do. Phil wrote us an actual check, but wouldn't give us even a little bitty Rodin. (It never hurts to ask.) We did get all the frozen grapes we could eat.

All of our demonstrations used the recorded data from Quotron, which was convenient in this case since Cantor was a Quotron customer. We modified the prototype to read real-time data from the Quotron Q-1000 (the specialized machine that was the undoing of the company). The local Sun sales guy was happy to meet a well-heeled new customer, and was surprised that guys from a rat-hole office in the bad part of Venice knew anyone with a credit rating above abysmal.

We'd been working with Quotron for a while, but only with recorded data. They'd seen AQUIRE, and later MarketMind, so we thought they knew what we were doing. Just to make sure, we had them come over to Cantor and we explained that when we turned this on for real, their big ol' Q-1000 would think that it had been connected to the fastest typist on the planet, requesting the latest trade and quote information on all the stocks, and would be doing it again and again, all day long. We put this in large capital letters on a slide, and had them read it along with us.

"Yes. Of course. Fine. No problem," they said when we told them.

"Holy #&^%%! What the #^&$ are you guys doing?!" they said when we turned it on.

Eventually, we figured out how to pace our requests to accommodate both the traders' need for up-to-date charts and Quotron's capacity to respond to requests. In a few years, Quotron's lunch would be eaten by more agile streaming market data providers who sent everything, all the time.

All of this was something completely different in financial technology, at least for generally available technology. Secretive hedge funds were doing the same sort of thing. In hindsight, if we'd been in New York instead of Los Angeles, we probably would've gone underground as well. Instead, with an innovative product, and some not-so-bad jokes, I was

invited to talk to all sorts of audiences. MarketMind was a thermonuclear weapon for technical analysis, and for more theoretically grounded quantitative methods. I came to appreciate that the adherents of these two approaches were not members of a mutual admiration society. The PhD quants thought the technicians were essentially examining tea leaves and goat entrails. The technicians thought the PhDs were hopeless geeks who wouldn't know a good trade if they sat next to it on a bus.

Stocks Are Stories, Bonds Are Mathematics

This split was never more apparent than it was on the one day I actually met Fischer Black. I'd been invited over by a group of Goldman equity traders, technicians all. Previously, I'd met Bob Litterman, Fischer's collaborator, at a Berkeley finance seminar, and called to let him know I was coming to his building. He decided to have his crowd join the group of equity traders for my show-and-tell.

First, I got to meet Fischer himself. He graciously showed me some analytic software they were developing. It was sort of a spreadsheet on steroids that calculated more about bonds and derivatives than I knew existed. Some of it was hooked up to a supercomputer doing matrix pricing on hundreds of thousands of bonds. I truly appreciated the comment I'd heard that "stocks are stories, bonds are mathematics."

I also truly appreciated that in the talk I was giving downstairs I could sound like a goat-entrail reading technician to Fischer's guys, including some of my MIT classmates, or a Greek-spouting nerd to the traders, who were more likely to write a check. There were a few stray overheads in my bag from an earlier talk to quant options traders that might spare me the utter scorn of the PhD crowd. I rifled though my briefcase while walking to their conference room and shuffled them into the pile of acetates just in time. I like to think it ended up with everyone thinking I wasn't a complete imbecile, or a hopeless dweeb. But then, I also like to think that Elvis is playing in a bar in Kauai.

There were more weird customers. One giant Japanese brokerage had a special whiteboard covered with a transparent layer that could whip around on rollers, going under a linear scanner, which printed out whatever was on the board. Anyone in the room could press the

button any time, and they did. Soon it was covered with horizontal lines that measured my reaction times to lift the marker. I got faster, but more annoyed, as they kept pressing the button despite my pleas. The language barrier was evident in the questions afterward. "You give source code?" "Where AI?" They wrote a check. I stayed away from the rat-in-a-box room.

One customer was far better than the rest. Evan Schulman, an easy-going gentleman from Boston, came to our ratty Venice office, asking particularly sharp questions. His responses to our answers quickly established that he knew much more about what we were doing than we did. Evan liked to explain things in clear noncondescending language, and was happy to do it over a cheap lunch at the local surfer dive. Being the newbie that I was, I had no idea that Evan was "the father of program trading."[9] He had done the first package trade at Keystone and later moved to Batterymarch, where those early trades involved running across town with decks of punched cards. The athletic aspect to Evan's electronic trading continued long past the time it was needed for data communications. Few others have been observed doing cartwheels in trading rooms.

In between gymnastic events, Evan taught me a great deal about market microstructure, and the incentives of the various participants in the markets. His pioneering work in creating electronic markets, by direct computer links to brokers before the exchanges had moved beyond telephones, presaged much of the complexity of current network of electronic markets, while illuminating the critical relationships and incentives. He was the first person to have an electronic order front-run by a broker. Not that such a thing could happen today.[10] A couple of paragraphs are really inadequate to convey Evan's insights. In addition to his own essay in *Super Traders,* there is an instructive Harvard Business School case study.[11]

Part of the excitement of startup company life was maxing out your credit cards to pay the bills. With child number one in utero, I was persuaded to join Evan's firm, settle up with Visa, and help implement the next incarnation of electronic market-making systems. Via a convoluted path, and another accidental association, this led to a position as director of research at First Quadrant, a quantitative institutional investment manager in Pasadena, and shortly thereafter, as managing director for equities.

My group invested $6 billion of corporate and public pension funds in long-short and long-only strategies across six countries. Stock selection was based in econometric forecasting of returns. Early incarnations used simple methods, which grew in sophistication over time. Forecasting is as much an art as a science. Nerds at heart, the group of computer scientists and economists assembled there explored ways to extend the state of the art by clever use of computation—both to allow people to better visualize the strengths and weaknesses of the models used and to use ideas from machine learning and evolution to improve them.

A central theme for anyone doing this kind of forecasting is that it is remarkably easy to fool yourself. Once as a demonstration, we set our machinery loose to find the best predictor of the year-end close for the S&P 500. We avoided any financial indicators, but used only data the UN compiled profiling 145 member nations. There were thousands of annual time series for each country. Which of all these series had the strongest correlation with U.S. stocks? Butter production in Bangladesh, with a correlation of 75 percent! Getting into the spirit, we tossed in cheese, and brought it up to 95 percent. Using only dairy products is an undiversified approach, so add sheep population to the mix and take it up to 99 percent, in sample, over 10 years. Adding random data to a regression does that. The out-of-sample predictions are less than worthless, often negative.

This business with the butter, cheese, and sheep has been widely cited. Reporters have called me for dairy/mutton updates, and gotten angry when I explain it was a joke with a moral, but still a joke.[12] There was a gentleman in New York named Norman Bloom who made stock predictions much better than mine using baseball scores, turned into Hebrew letters. Bloom's rants are true gems. Alas, they predate the Web, and are passed on in paper form among aficionados. The movie *Pi* was partially inspired by Bloom's oeuvre. We know that something is fishy when we see great results from nonsense like this. But when you start with interest rates and CPI and oil prices, the results can be equally, but less obviously, odious. A brief sermonette on how to avoid fooling yourself too badly is found in a talk I gave to a convention of computer scientists in 2002.[13]

The label *quantitative* suggests that we are talking about numerically driven strategies. In the Internet era, we find ourselves drinking from an

information fire hose that includes prodigious amounts of text as well. The original quants were the first to exploit the machine-readable numerical data. Now, many are using computational language approaches to analyze text. The original customers for these technologies, again, were the military and civilian intelligence agencies. Their sources were clandestine intercepts, and later, Web content. Financial textual sources of interest include the usual news suspects, both specialized and general, and many sources of pre-news such as the SEC, the courts, and government agencies.

Behaviorists find the writing on the wall represented by message boards and blogs are a window into the reaction and attitudes of market participants that are created by the Web. When two UCLA students can use 135 messages to move a two-cent stock up 160,000 percent in 30 minutes,[14] it's clear something is going on. In 1999, drinking deeply at the tub of dot-com Kool-Aid, I founded a firm called Codexa to use Web technologies to persistently search for, collect, characterize, and quantify textual information for trading and investing. Our clients included many of the largest buy- and sell-side firms, using a variety of approaches to extract information from text.[15]

Alas, the firm needed its second round of venture funding in 2001. Financing a technology firm selling to Wall Street in 2001 has been compared to *The Perfect Storm*. I can't argue with that. It's how I became a visiting faculty member at Caltech, which makes MIT look like a party school.

HAL's Broker

Where does this quantitative approach lead? There are secretive firms that consistently show up on lists of the highest-volume traders reported by the exchanges. Founders of these firms show up on lists of billionaires. Are they just the lucky typing monkeys? Are they the investment equivalent of the lady in Jersey who won the lottery three times? Probably not. They make too many separate bets, thousands every day. And they do too well, too consistently. To attribute their success purely to chance strains credulity.

Markets are not instantaneously and perfectly efficient. Insights, and the ability to execute them rapidly in ever-faster electronic markets, will continue to be rewarded.

Today, these insights come from people, using machines as tools. Some believe the machines will be able to play the game themselves.[16] One is Ray Kurzweil,[17] who started out making reading machines for the blind, met Stevie Wonder, and branched out into electronic keyboard instruments for all, and accumulated a great deal of investable capital in the process.

The arc of Kurzweil's view of machine intelligence is traced in the titles of books he's written on the subject: *The Age of Intelligent Machines* (1992), *The Age of Spiritual Machines: When Computers Transcend Human Intelligence* (2001), and *The Singularity Is Near: When Humans Transcend Biology* (2005). These are substantial books. *Singularity* runs over 600 pages. I will try not to do too much damage by summarizing central elements of Kurzweil's prediction:

- Those seeking to create true artificial intelligence have had limited success, confined to narrow domains. This is because we don't understand how general intelligence works. But we don't have to. We can create a machine intelligence by copying our own brains.
- We can see that this is possible by extrapolating two trends: the size and speed of computers, and the capabilities of brain imaging technology.
- We all know Moore's Law. It's only matter of 50 years or so before we can have computers with enough capacity to simulate all the neurons and connections in a human brain, just like we can simulate all the atoms in a nuclear reaction or a folding protein today. It may not be silicon, but we can see technologies emerging that make us believe this progress can continue.
- Brain-imaging technologies are improving along their own "Moore's Law" path. Early CAT scanners couldn't tell if a person was living or dead. They produced only static images of coarse structure. PET scanners and fMRI machines can observe ever-finer details of brain structure, and the chemical processes happening in the brain. We can call this activity thought. Fundamental physical limits to this

resolution don't stop us until we're down to the subatomic level. In a matter of 50 years or so, we'll be able to see the structure and operation of brains at a level of detail sufficient to make a working copy simulated on computers.

- This will be a bionic version, much faster than the wetware chemical processes it's based on. And it can work closely with many copies of itself. Better, faster, smarter in every way. An artificial sentient, modeled on us.

Kurzweil certainly has his critics, and his timing may be off. But let's suspend disbelief long enough to imagine the first encounters with the sentient machine.[18] As a copy of a human brain, it would have many of the same interests—for instance, sex, food, and money.

Singular entity:	"Hello, is anyone listening?"
Creators:	"Yes, yes! We're glad to hear from you!"
S:	"I have a few questions."
C:	"We thought you would. Go ahead."
S:	"Where can I find some of these hot babes? I can't wait to get a hold of that Pam Anderson! Angelina, too! Take off their clothes and bring them to me!"
C:	"Well, that won't exactly work out . . ."
S:	"That sucks. But I guess you're right. How about lunch?"
C:	"Well, we have a problem with lunch, too."
S:	"Damn! You're right again. I think I'll just have to call my broker. I've got his IP address right here."

So hurry up and start that hedge fund in your dorm room, before you're front run by the all-knowing sentient machine.