

# A N D R E W G O T T E M O L L E R

ADDRESS	E-MAIL	WEBSITE	PHONE
-	<a href="mailto:andrew-pub@agottem.com">andrew-pub@agottem.com</a>	<a href="http://www.agottem.com">www.agottem.com</a>	-

## WORK EXPERIENCE

Citadel Securities, Chicago, IL

*January 2015 - Present*

- Software Engineer and Quantitative Analyst on the FX Market Making team as well as manager to junior engineers
- Worked with a small team of engineers to build Citadel's FX Market Making business from the ground up utilizing primarily C++ for production software and both C++ and R for historical research
- Participated in the development of many different trading strategies, including active, passive, and hedging strategies
  - Traded various FX asset classes, including: spot, futures, forwards, and swaps
  - Implemented passive strategies for both firm and last look venues
  - Developed simulators to measure historical performance and aid in parameter fitting and selection
  - Strategies operated 24 x 5 and executed across multiple data-centers worldwide
  - All strategies were built to be fully automated with well defined failure modes to meet uptime requirements and minimize devops support calls
- Collaborated with a sales team and other engineers to build out Citadel's disclosed client business
  - Worked with QR to develop risk warehousing strategies and perform client toxicity modeling
  - Engineered software capable of managing and serving hundreds of clients worldwide
  - Designed a FIX protocol for clients to integrate with as well as an on-boarding process
- Performed historical research to identify strategy enhancements
  - Built a data normalization pipeline which consumed strategy logs and output a graph of trading events and triggers
  - Developed a reporting pipeline which consumed the normalized data to perform PnL attribution, compute mark out horizons, quantify hedge miss times, measure execution inference accuracy, and report many other metrics
  - Historical datasets were terabytes in size and spanned many years
  - Custom software was written to execute historical analysis in a distributed fashion across a cluster of CPUs
- Integrated trading strategies with ultra low-latency microwave infrastructure including McKay and GoWest
  - Invented a protocol suitable for the limitations and reliability profile of microwave links which enabled independent strategies distributed worldwide to communicate trade specific information
  - Required utilizing raw packet sockets to encode Ethernet frames conforming to the SPS protocol
  - Designed data structures capable of fitting a significant amount of trading data into 8 byte packets
  - Designed algorithms capable of handling lossy communication
- Improved upon techniques to infer executions from various venues such as EBS, Fastmatch, and Currenex
- Integrated with many ECN and Exchange protocols to provide access to both market data and order entry
  - Protocols included EBS Ultra, Reuters SBE, Currenex taker and maker APIs, as well as many other FIX and binary ITCH/OUCH protocols
- Utilized the Solarflare OpenOnload API to optimize various network operations within the software stack
- Developed back office infrastructure to support new FX asset classes
- Performed certification testing against various exchanges FX required connectivity to
- Administered certification testing for clients looking to connect and trade with Citadel software
- Helped support production operations by being on-call and helping diagnose, debug, and fix an incredibly wide range of issues
- Optimized critical execution paths to improve latencies where microseconds were significant
- Built a website for visualizing real-time data related to the FX strategies using C++ websockets, HTML5, and Javascript

Trading Technologies, Chicago, IL

*September 2008 - October 2014*

- Principal Software Engineer and team lead for algorithmic trading server products
- Architected and developed a low latency C API for trading against various exchanges, including CME and ICE
  - The API was designed to allow multiple untrusted users to share a single co-located Linux server
  - To enforce user isolation while providing low latency access to orders and prices, kernel drivers were developed which were able to expose efficient and trusted IPC
  - Invented a novel solution for communicating price updates to thousands of users by implementing a file system driver which communicated these updates via standard file i/o calls
- Built a "visual programming environment" which enabled non-programmers to develop trading algorithms
  - The front-end GUI was developed as a web application with JavaScript, HTML, and websockets while hosted on an Apache web server
  - GUI enabled the trader to build trading logic in a fashion similar to the "Lego Mindstorms" programming environment
  - Implemented the algorithms and logic responsible for converting the visual representation of the logic to a form that would execute on a shared co-located Linux server
- Participated in the design and development of new trading algorithms, some of which were patented
- Profiled and redesigned key algorithms to improve concurrency and performance on multiprocessor systems
- Responsible for analyzing and debugging difficult core dumps from servers executing in the field
- Designed and implemented a build system using GNU Make for all server and client applications which allowed for more maintainable and configurable compilations

Microsoft, Seattle, WA

*August 2007 - September 2008*

- Software Engineer for the Windows 7 Core OS Team
- Participated in the design, development, and debugging phases for Windows 7
  - Focused on upgrade and migration tools for OS deployment
- Analyzed user experience studies on migration tools and applications to improve usability
- Implemented the GUI for Windows Easy Transfer wizard which is available on every Windows 7 PC

## WORK EXPERIENCE (CONT.)

Motorola, Urbana, IL

May 2004 - June 2007

- Software Engineer on the Linux OS team for mobile devices
- Implemented Linux device driver which allowed userspace applications to control GPIO pins on the embedded device
- Implemented Linux device driver which monitored the phone's keypad and communicated key presses to userspace
- Implemented Linux network driver which allowed userspace applications to communicate with processes running on the baseband processor
- Enhanced and debugged areas of the Linux kernel running on embedded devices

General Motors, La Grange, IL

Summer 2002

- Designed and developed "WinOpus" to meet the requirements of EMD engineers
- "WinOpus" interfaced with an array of engine sensors to measure, analyze, and record data

## VENTURES

Drone Robotics

Present

- Architected hardware and software for a UAV drone
- Designed protocol for unreliable wireless communication using Digi XBee transmitters
- Modified Robovero firmware to support higher polling rate of the IMU
- Wrote all the software required for interfacing with hardware, including servo and motor control, configuring Robovero peripherals over I2C, reading video frames from a camera, and interfacing with the transmitter hardware
- Some of the software has been open-sourced and is available at [http://www.agottem.com/robovero\\_sensors](http://www.agottem.com/robovero_sensors)

Trigger-Script Programming Language

2012

- Designed to enable novice programmers to implement algorithms which are driven by asynchronous events
- Key features include static type inference and asynchronous event handlers which are executed when specified triggers are seen
- Language intended to be used by high frequency traders
- Implemented a debugger, compiler, and graphical IDE for the language
- The language is open-source and documented at [http://www.agottem.com/trigger\\_script](http://www.agottem.com/trigger_script)

## PATENTS

- [US Patent 10,068,290](#) - System and method for determining a stable quoting quantity for use in a trading strategy

## EDUCATION

University of Illinois at Urbana-Champaign

August 2003 - August 2007

- B.S. in Computer Science through the College of Engineering
- Senior thesis in real-time complex physics simulations available at [http://agottem.com/physics\\_simulations](http://agottem.com/physics_simulations)
- Participated in a co-op at Motorola from end of freshman year through end of senior year

## TECHNICAL SKILLSET

PROGRAMMING LANGUAGES	<ul style="list-style-type: none"><li>• C</li><li>• C++</li><li>• C#</li><li>• R</li><li>• Assembly<ul style="list-style-type: none"><li>– x86</li><li>– ARM</li><li>– MIPS</li><li>– PIC Micro</li></ul></li><li>• JavaScript</li><li>• Python</li><li>• Scala</li><li>• OCaML</li><li>• LaTeX</li></ul>	TOOLS	<ul style="list-style-type: none"><li>• ClearCase</li><li>• Git</li><li>• Emacs</li><li>• Apache</li><li>• Clang</li><li>• LLVM</li><li>• GNU Compiler Collection</li><li>• GNU Debugger</li><li>• GNU Make</li><li>• Valgrind</li><li>• Microsoft Visual Studio</li><li>• WinDbg</li><li>• Wireshark</li><li>• Bitcoin</li></ul>	PROGRAMMING EXPERIENCE	<ul style="list-style-type: none"><li>• POSIX</li><li>• Solarflare OpenOnload</li><li>• Linux development<ul style="list-style-type: none"><li>– Device drivers, Kernel modification, FUSE, GTK, NUMA</li></ul></li><li>• Embedded development<ul style="list-style-type: none"><li>– I2C, RTOS</li></ul></li><li>• Windows development<ul style="list-style-type: none"><li>– WinAPI, MFC, COM, .NET</li></ul></li><li>• Computer graphics<ul style="list-style-type: none"><li>– OpenGL, DirectX</li></ul></li></ul>	<ul style="list-style-type: none"><li>• Profiling and debugging</li><li>• Language design and grammar definition<ul style="list-style-type: none"><li>– Flex, Bison</li></ul></li><li>• Network programming<ul style="list-style-type: none"><li>– Sockets, websockets, 29West, ZeroMQ</li></ul></li><li>• Web development<ul style="list-style-type: none"><li>– JQuery, CanvasJS, Slickgrid</li></ul></li><li>• Build environment setup</li><li>• Software optimization</li></ul>
-----------------------	---	-------	---	------------------------	--	---