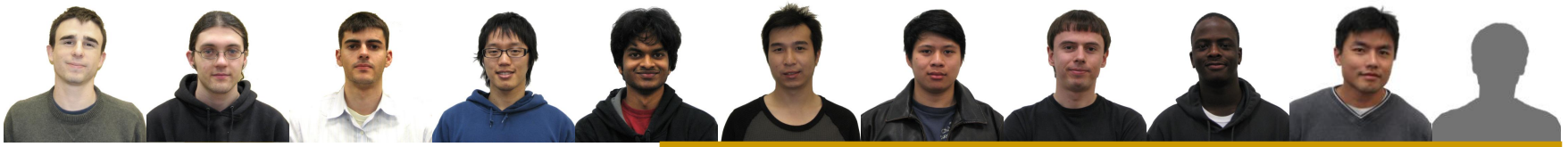
IPRO 313 2 MB/s

ULTRA-HIGHSPEED MARKET DATA TICKER SYSTEM



FINAL REPORT

Faculty Advisor
Wai Gen Yee
Ben Van Vliet

Team Members
Oluwaseun Shonubi
Khanh Duong
Jose Acuna-Rohter
Tarun Anupoju
Lance Cooper
Martin Kolodziej
Konstantin Roytman
Jing Kai Tan
Jong Su Yoon

Problems

- People had access to substantially larger volumes of data with significantly less latency
- To provide assistance to those seeking a tool that provides the most up-to-date financial data

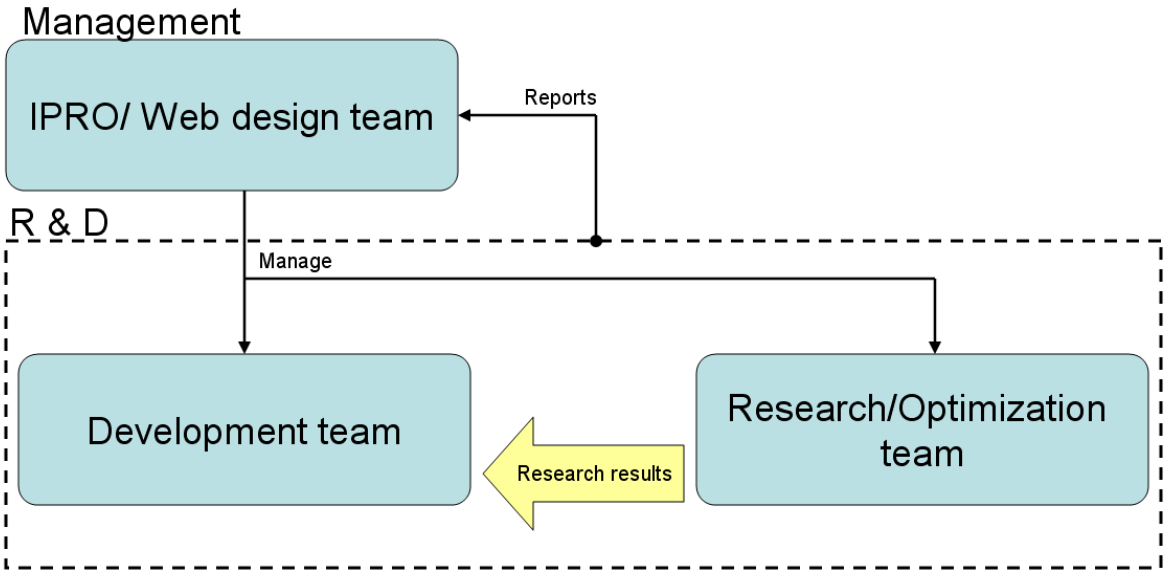
IPRO Goals

- To create a high performance data ticker system
  - proof-of-concept data ticker plant that processes real data by the end of the semester
- The learn to work as a team

Objectives

- Explore competitors' solutions and available technology
- Develop a functioning ticker plant system
- Improve system performance
- Determine hardware requirements
- Update the technical user manual
- Create a website

Team division



- Development team
  - Designing & Implementing the system
- Research/Optimization team
  - Research solutions to improve base system developed
- IPRO/Web Design team
  - Project management, IPRO deliverables
  - Project website

Development team

Developing the software and hardware for the system.
Responsible for design and implementing the system.

Jose Acuna-Rohter CS

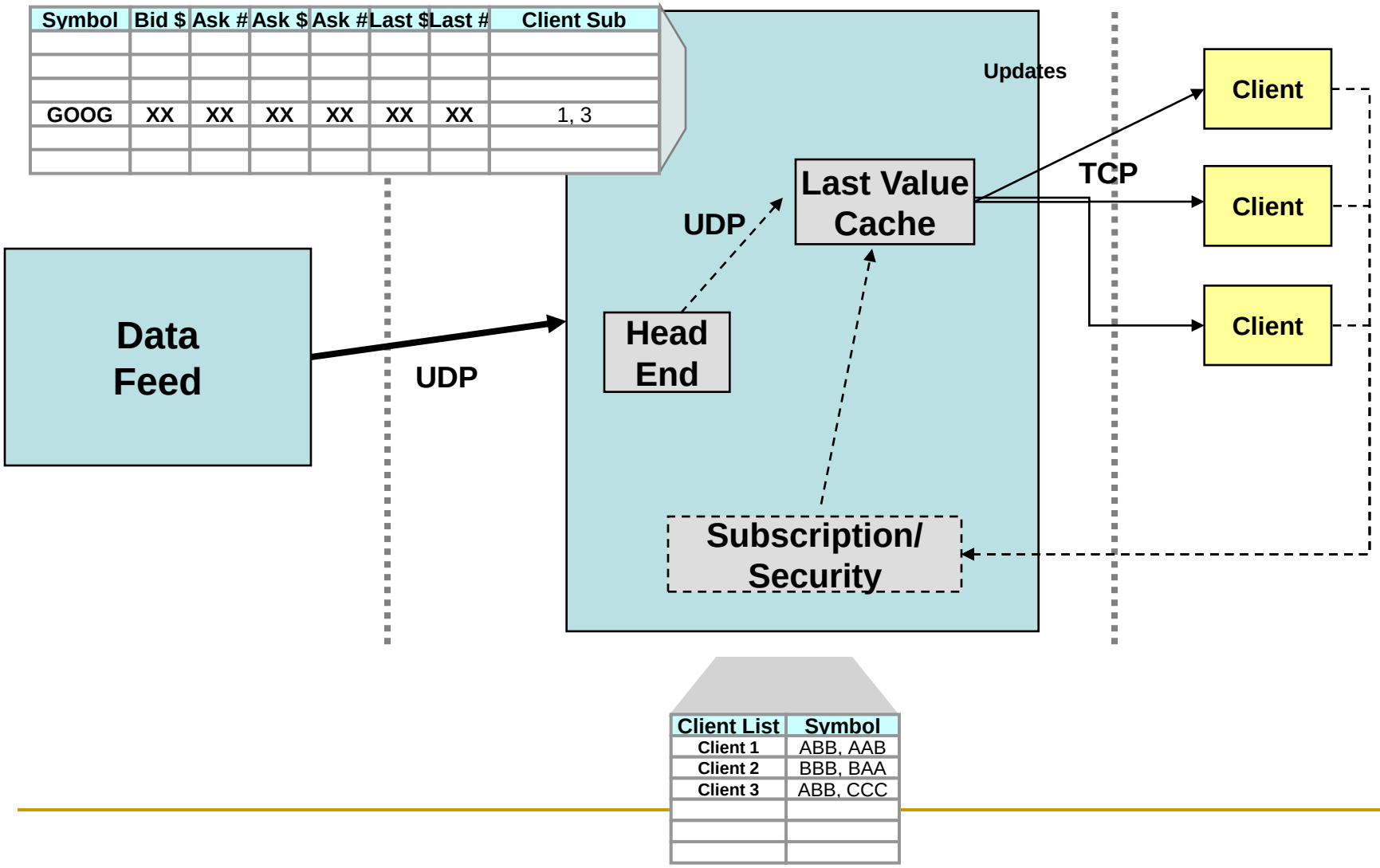
Jong Su Yoon CS

Konstantine Roytman CS

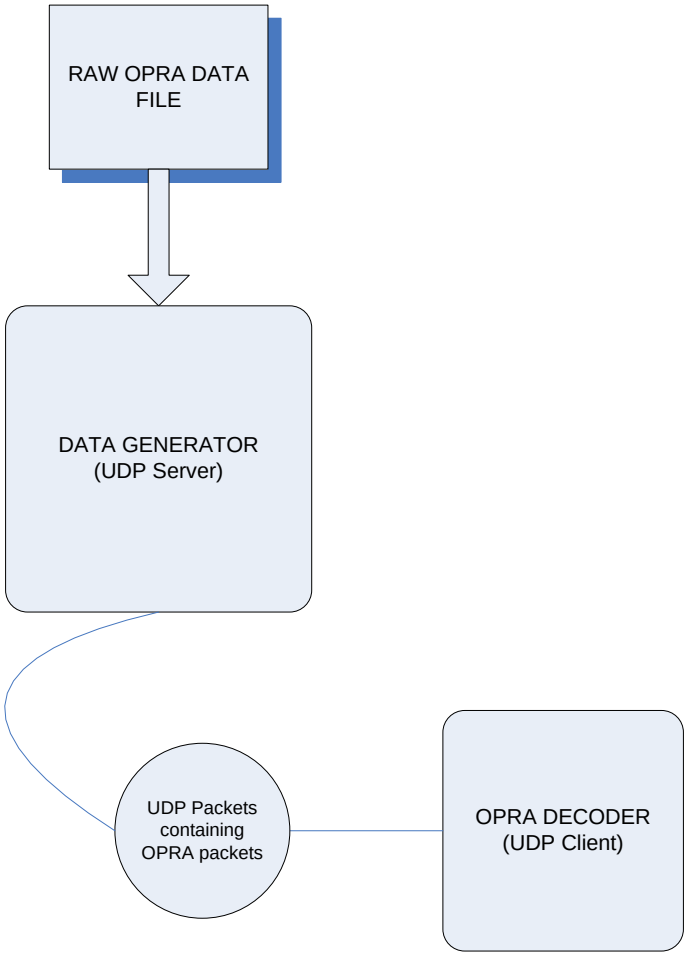
Lance Cooper CS

System Design

- Current Status



Data Feed


RAW OPRA DATA FILE
DATA GENERATOR (UDP Server)
UDP Packets containing OPRA packets
OPRA DECODER (UDP Client)

Head End - Inputs

- Receive OPRA FIX FAST encoded data
- Receive data via UDP
- Incoming Data is grouped in OPRA Packets
- Each OPRA packet must be completely decoded before more packets are read in.

Head End - Output

- 40 byte message is created
- Position Based Message
- Sent to Last Value Cache(LVC) via UDP
- 1 message is sent per UDP packet

Head End Message Structure

- Sample Message:
  - FVXWHB0004000A0000000100020000000 1100020

From	To	Field Description	Field Length
0	3	Symbol	3
3	4	Expiration Date	1
4	5	Strike Price Code	1
5	6	Strike Price Denominator Code	1
6	13	Explicit Strike Price	7
13	14	Premium Price Denominator Code	1
14	22	Bid Quote	8
22	27	Bid Size	5
27	35	Ask Quote	8
35	40	Ask Size	5

LVC

- Stores the last seen value for all symbols that pass through the system.
  - Updates must be stored rapidly to avoid loss
  - We use a hash table to do this efficiently
- Subscription submodule distributes updates to connected clients, and handles client requests to subscribe or unsubscribe to a symbol.

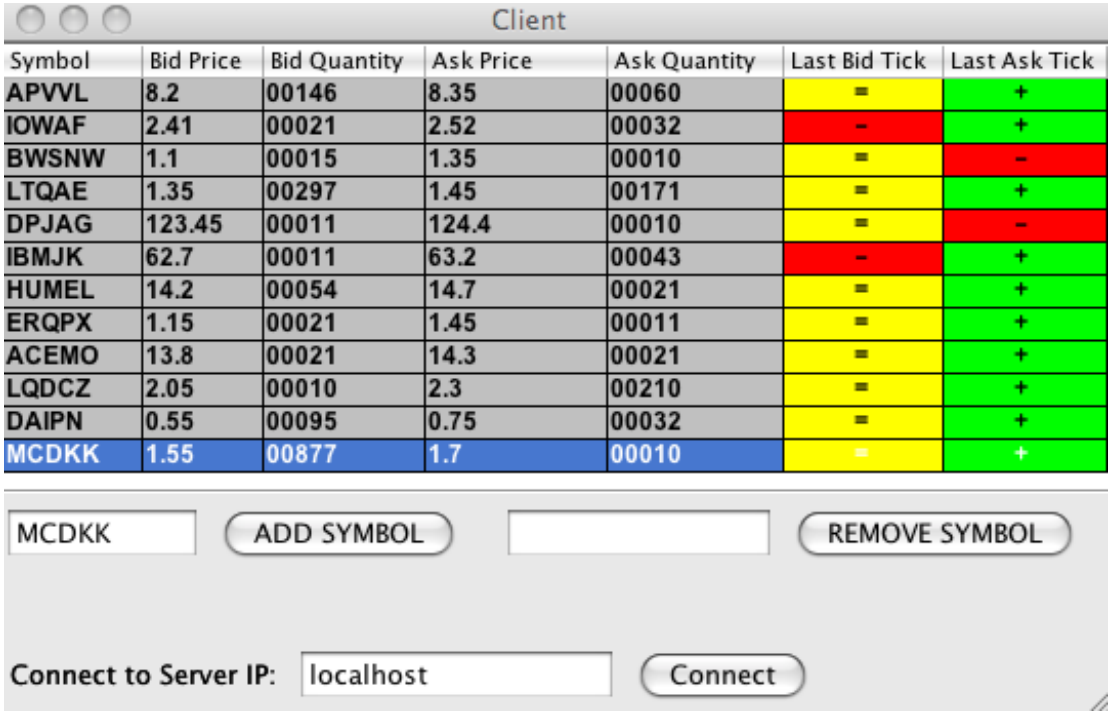
LVC: Hash Performance

- The most performance critical aspect of the LVC is the hash algorithm used.
  - Tested:
    - CRC32
    - STL
    - Jenkins
    - AlphaNum
  - The established CRC32 proved performed well enough to continue use, but AlphaNum showed promise warranting investigation.

Client

- The client program allows user to get price update from the server on selected symbols.
- Function
  - You can make a connection to LVC
  - You can add/remove symbol through add/remove symbol button.

Client



It shows you change of the price with color

- Yellow: no change.
- Green: increasing
- Red: decreasing

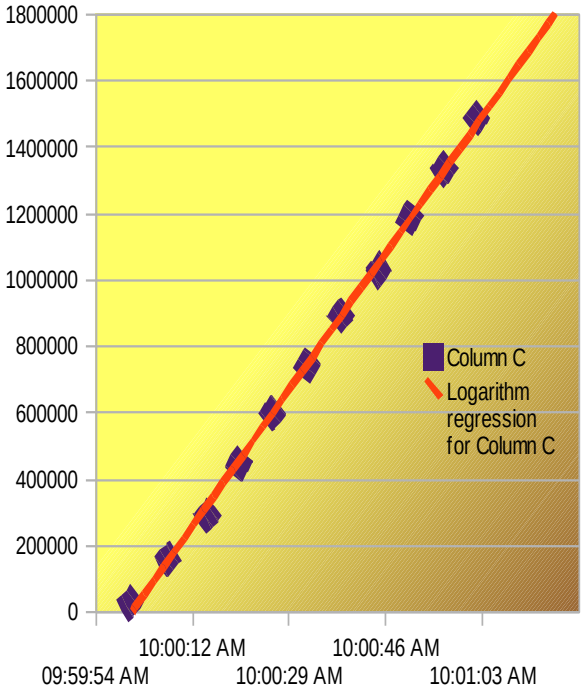
Research/Optimization Team

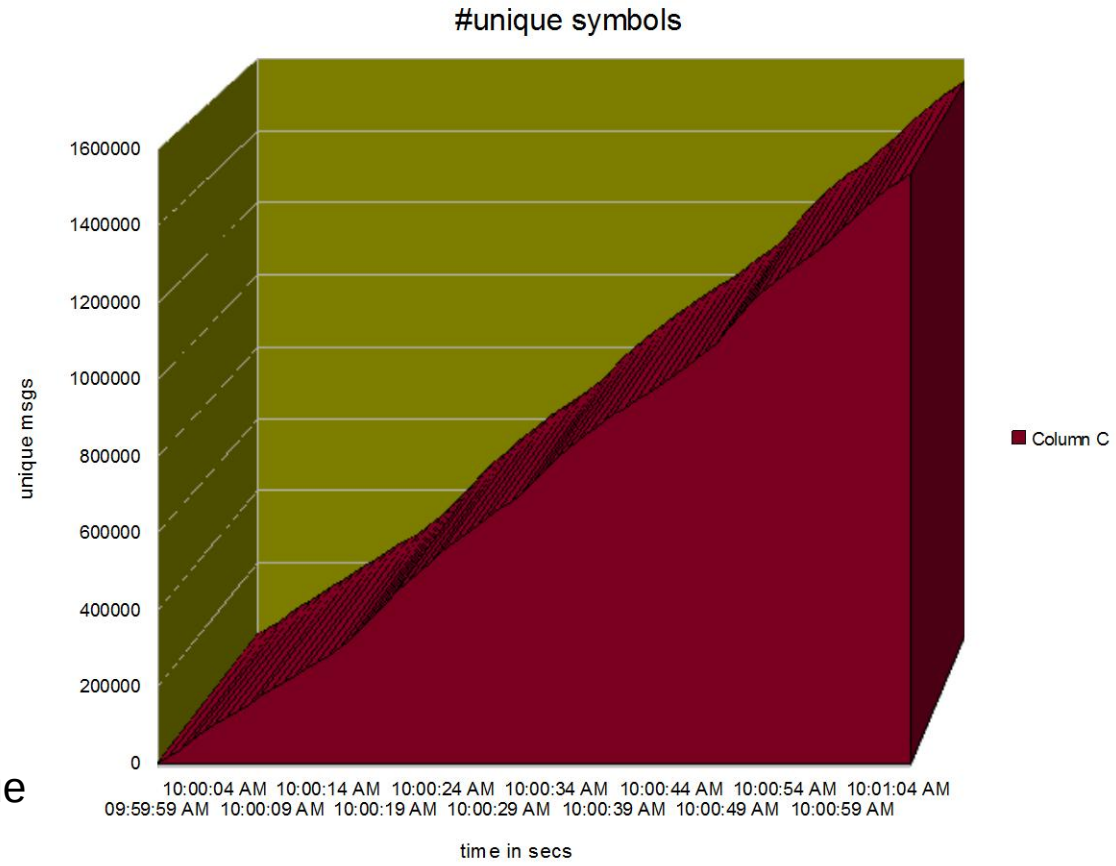
Look at the base system developed, research for solutions to improve and optimize the system for maximum reliability and performance.

Tarun Anupoju CPE **Jing Kai Tan** EE

Analysis of Real OPRA Data

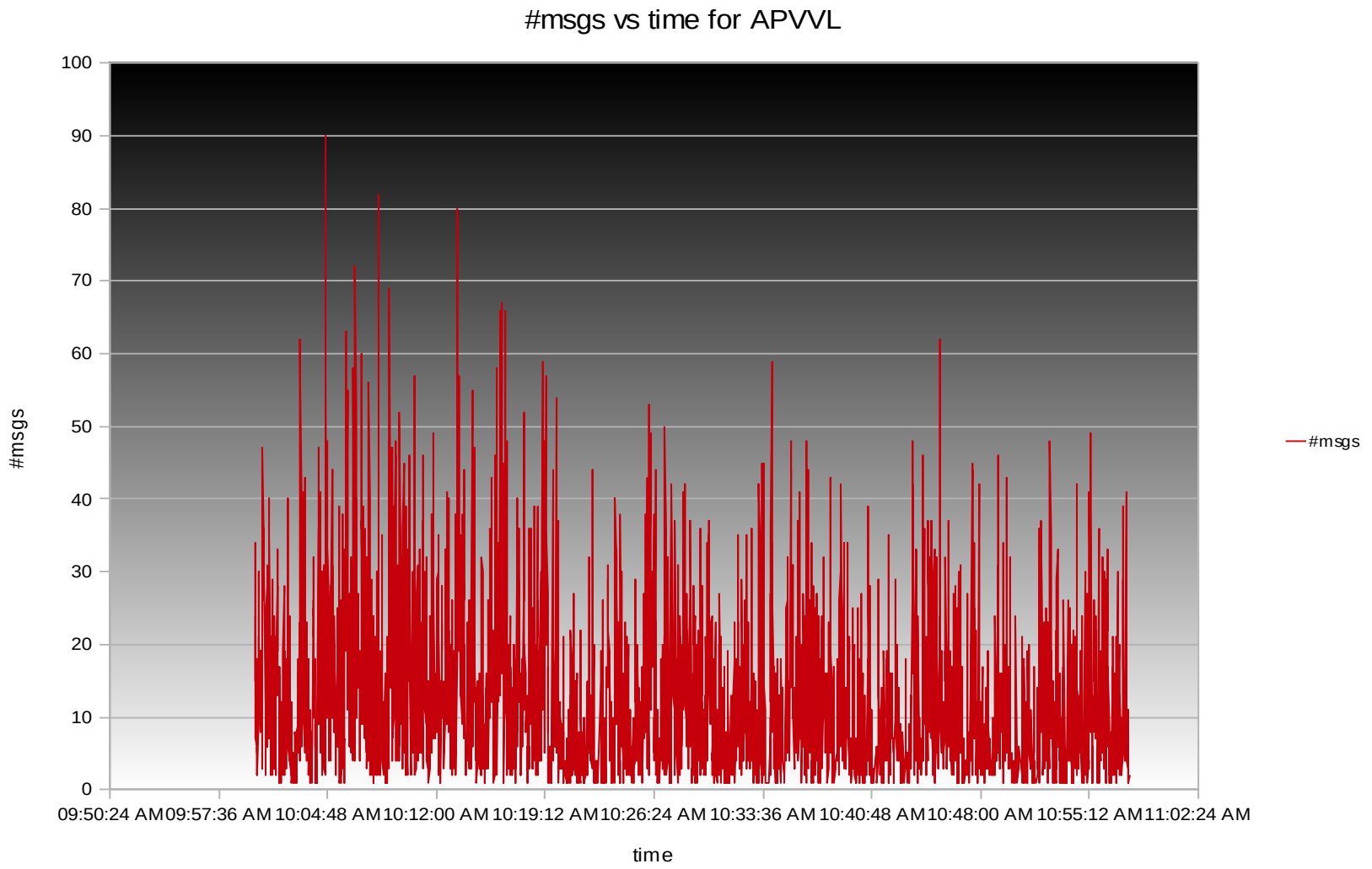
- 2 real OPRA data sets. OPRA9.OUT and OPRA10.OUT
- Contains data between 9am and 11am
- Analysis gives understanding of what the system is handling.





Graph to the right: Show the unique symbols in the 1st 60 seconds

- Results form the basis for the requirements of the Hash Function

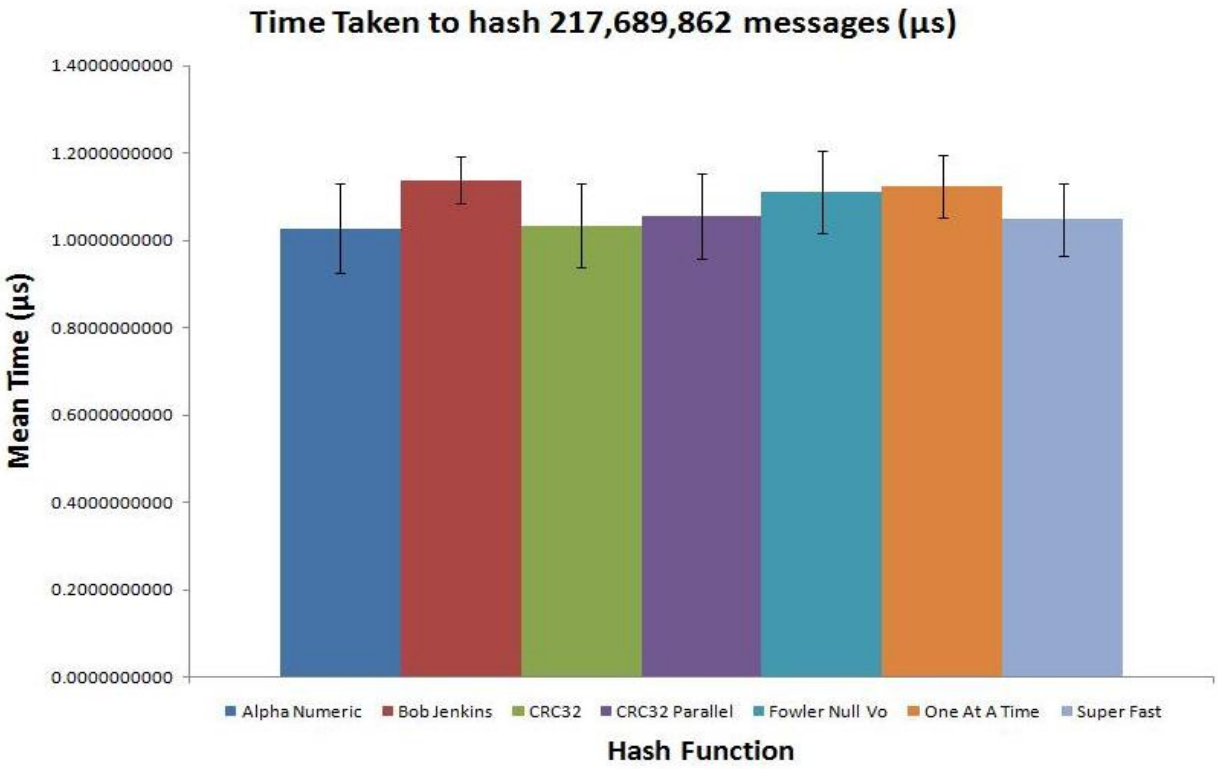


Graph shows the option put symbol APVVL which belongs to Apple Inc

Hashing

- Hash Function tested:
  - Alpha Numeric
  - CRC32
  - Fowler Noll Vo
  - Super Fast
  - Bob Jenkins
  - CRC32 Parallel
  - One at a Time
- Testing Platform:
  - Intel Centrino 2.50 GHz
  - 2.0 GB Memory Ram
- Measurement:
  - Time taken to hash 217 millions of OPRA Symbol (5 Characters)
  - Number of collision in 113,263 unique hash key

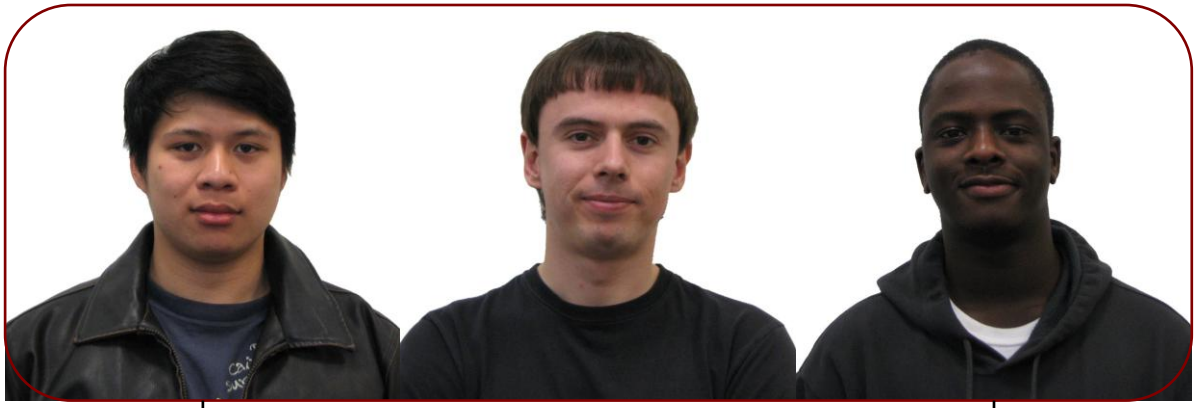
Hash Result



Hash Function	Alpha Numeric	Bob Jenkins	CRC32	CRC32 Parallel	Fowler Null Vo	One At A Time	Super Fast
Average Time (µs)	0.4776880524	0.5155182644	0.5014418247	0.4981016012	0.5270051974	0.5144626931	0.4981157233
Number of Collision	0	3	0	183	0	13	899

IPRO/Web Design team

Responsible for handling the creation of IPRO deliverables and the IPRO 313 Website



Martin Kolodziej EE

Khanh Duong CPE

Oluwaseun Shonubi EE

IPRO 312 website

- phpWiki driven design
- Support for Firefox and IE 7
- Includes web client
- Integrated search option
- http://www.iit.edu/~ipro313s08/

IPRO 313
ULTRA-HIGHSPEED MARKET DATA TICKER SYSTEM

IIT

HOME ABOUT DOCUMENTS WEB CLIENT LINKS

Introduction - Objective - Team Structure - Challenges - Updates

Updates:
- Web client added
- Team structure
- Documents
- Links

Progress

Development
100%

Research
100%

IPRO Deliverables
85%

Website
80%

Title search:
Search

Full-text search:
Search

Ultra-High Speed Market Data Ticker System

With the dawn of every New Year, the speed of business is ever increasing. What used to be performed in months, days or minutes is now being done in milliseconds. To stay competitive within their industries, businesses today, particularly in finance we need real time access to large volumes of data. This requires vast improvement in information technology infrastructures.
The objective of IPRO 313 is to create a high performance data ticker plant. Ideally, the data ticker plant has to have a sustained optimal throughput of three million price quotes per second &#65533; the current industry state of the art - with minimal latency. Toward this end, the team will create a proof-of-concept data ticker plant that processes real data. The initial system will be used as a baseline on which optimizations to specific components will be made.

Introduction of Electronic Trading

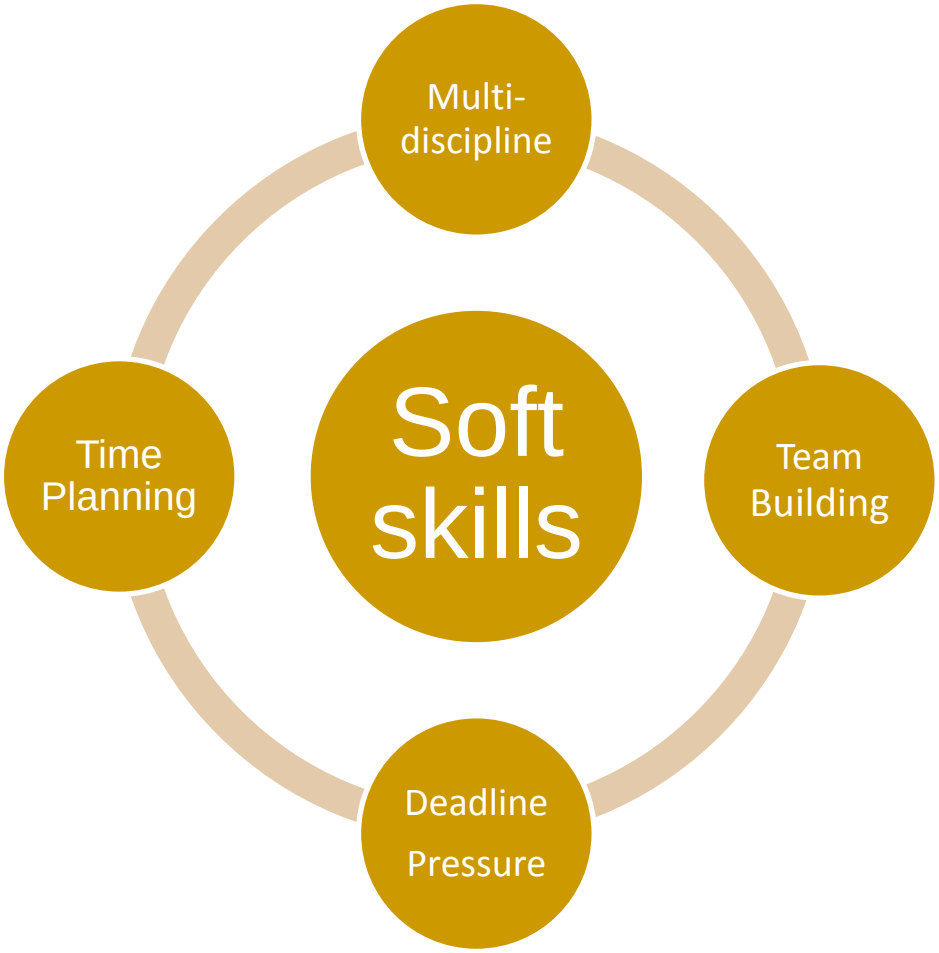
The 1975 Security Acts Amendment, enacted by Congress in 1975 gave the SEC the authority to create an efficient national market infrastructure (read electronic trading technologies). The reason for this was to eliminate as much of the redundant paper work that slowed the trading process down as possible. As a result, people had access to substantially larger volumes of data with significantly less latency.

Obstacles and Other Team Related Issues.

SOFT SKILLS

- Difference in disciplines of the respective team members
- Time management
- Team building / individual motivation
- Meeting deadlines set by fellow team-mates and laid down deadlines from IPRO office.

Visualization of Obstacles



Conclusions

- We fixed the system from what it previously was
- We created a base system.
- We followed the rules/guidelines from the IPRO office.
- IPRO was a success!
- Questions?