Federal Market Information Technology in the Post Flash Crash Era: Roles for Supercomputing

E. Wes Bethel
David Leinweber
Oliver Rübel
Kesheng Wu

Lawrence Berkeley National Laboratory
One Cyclotron Road
Berkeley, CA 94720

DISCLAIMER

This document was prepared as an account of work sponsored by the United States Government. While this document is believed to contain correct information, neither the United States Government nor any agency thereof, nor the Regents of the University of California, nor any of their employees, makes any warranty, express or implied, or assumes any legal responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by its trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof, or the Regents of the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof or the Regents of the University of California.

Federal Market Information Technology in the Post Flash Crash Era: Roles for Supercomputing

E. Wes Bethel, David Leinweber, Oliver Rübel, Kesheng Wu
(authors in alphabetical order)

September 16, 2011

Abstract

This paper describes collaborative work between active traders, regulators, economists, and supercomputing researchers to replicate and extend investigations of the Flash Crash and other market anomalies in a National Laboratory HPC environment.

Our work suggests that supercomputing tools and methods will be valuable to market regulators in achieving the goal of market safety, stability, and security. Research results using high frequency data and analytics are described, and directions for future development are discussed.

Currently the key mechanism for preventing catastrophic market action are “circuit breakers.” We believe a more graduated approach, similar to the “yellow light” approach in motorsports to slow down traffic, might be a better way to achieve the same goal. To enable this objective, we study a number of indicators that could foresee hazards in market conditions and explore options to confirm such predictions. Our tests confirm that Volume Synchronized Probability of Informed Trading (VPIN) and a version of volume Herfindahl-Hirschman Index (HHI) for measuring market fragmentation can indeed give strong signals ahead of the Flash Crash event on May 6 2010. This is a preliminary step toward a full-fledged early-warning system for unusual market conditions.

Keywords: Flash crash, liquidity, flow toxicity, market microstructure, probability of informed trading, VPIN, circuit breakers, market fragmentation. (JEL codes: C02, D52, D53, G14.)

1 Introduction

After the dramatic Flash Crash of May 6, 2010, it took more than four months in order for the SEC/CFTC to issue a full report on the event [4, 5]. Such a long duration of time was required because the government currently relies on a mélange of legacy systems. The SEC and CFTC clearly realized the limitations, and have called for comments in a 203-page discussion of the proposed next-generation system: CATS (Consolidated Audit Trail System) [25]. In computer science, visionary scientists like Jim Gray have developed technologies and tools to foster a brand new approach termed “data-intensive science [13]”. This paper reports our initial attempt to adapt some key techniques to address the information technology needs of financial regulatory agencies.

A basic tool in regulating the financial market is the “circuit breaker” that stops trading. However, many traders and academicians have compared this approach of on/off circuit breakers to “applying the rules of the road to aircraft.” As different markets and venues become more interdependent, sudden halts in one market segment can ripple into others [16, 28]. One key idea is that brown-outs are preferable to black-outs. After the Flash Crash of 2010, new circuit breakers were instituted that stop the trading of individual stocks if their price variations exceed a prescribed threshold [1]. However, this approach is insufficient for achieving safety and stability. It may be possible to detect and predict hazardous conditions in real-time, allowing the regulatory agencies to implement a "yellow light" approach, to slow down, rather than stop markets.

In this work, we seek to explore three aspects of implementing these early warning indicators: i) finding indicators that can provide early warnings, ii) better understanding of their computational requirements, and iii) implementing an interactive exploratory system for validation of warning indicators and to allow expert verification in cases requiring extraordinary actions.

Based on recommendations from large traders, regulators, and academicians, we have implemented two sets of indicators from institutional traders that have been shown to have "early warning" properties preceding the Flash Crash. They are the Volume Synchronized Probability of Informed Trading [10] (VPIN) and a variant of Herfindahl-Hirschman Index (HHI) of market fragmentation [12, 14]. We describe how to organize data to efficiently compute these indicators and how to parallelize the computational tasks on a high performance computer (HPC) system. Because the automated indicators may need to be verified if a critical action is to be taken, we also explore the option of interactively verifying warnings with comparisons to historical data by using a high-performance search and visualization tool. One major operational challenge in the future will be to perform all these tasks – including computation and verification of market warnings – in real-time.

The two early warning indicators use trades only, and the interactive exploration system uses trades and quotes, neither of them uses the limit order book or order flow data, so they are modest examples of what would be required in an operational system. Nevertheless, we believe this exercise is instructive because the basic approach could be extended to work with higher levels of data. Furthermore, this exercise enables us to identify critical issues to be addressed in order to build a high-performance monitoring system for regulators responsible for the safety, security, and stability of financial markets.

2 Background

2.1 LBNL Center for Innovative Financial Technology

In 2010, Horst Simon, then director of LBNL's Computational Research Division (now deputy director of LBNL) and David Leinweber, a computer scientist, financial technology entrepreneur [19] and Fellow in Finance at the Haas business school co-founded LBNL's Center for Innovative Financial Technology (CIFT). In addition to involving senior HPC researchers in this effort [17], we also have assembled a multidisciplinary collaboration of the type mentioned in the call for papers of the Workshop on High Performance Computational Finance at SC11.

This group has engaged with key federal regulators at the SEC and Treasury [3], high frequency traders, financial data and tool vendors, and leading academics in market micro-structure. CIFT's initial approach was to replicate portions of the Federal Flash Crash investigation to gain firsthand experience into how HPC technologies, e.g., FastBit [31], are applicable in market analysis. More recently, we have tried to follow the examples set by Jim Gray and other key figures of data-intensive sciences [13]. In particular, the Flash Crash of 2010 revealed an urgent need for the financial regulatory agencies to apply the modern data management and analysis techniques to improve market safety, security, and stability. A call for action along these lines appeared on page 1 of the spring issue of the Journal of Portfolio Management, which is one of the most prestigious publications in the financial industry [18].

To realize the above vision, CIFT is collaborating closely (at code level) with the developers of VPIN to evaluate its effectiveness as an early warning indicator of flash crashes and other market anomalies [10]. We have also taken a similar approach to implement and evaluate a family of fragmentation indices suggested by the head of trading research at one of the world's largest investment firms (assets under management of $3.6 trillion) [20]. These collaborations are supported by proven HPC technologies for searching and visual analyses [6, 30], along with the computing power of one of the most powerful supercomputers at the National Energy Research Scientific Computing (NERSC) center [1].

2.2 Levels of High-frequency Financial and Market Data

In this section, we briefly describe the types of data used in our work. Readers are referred to texts on market microstructure for a full treatment [23].

2.2.1 Level 1 – Transactions

The most basic form of market data is the most familiar – a time series of prices, widely seen on smartphones and market television, and screens used by millions. Popular time series, such as the Dow-Jones Industrial Average (DJIA), are weighted averages of stock trades.

2.2.2 Level 2 – Transactions and BBO Quotes

All transactions from Level 1 data, plus the "inside quotes," i.e., the best bid and offer price for each security in the combined markets included in the data. The Trades And Quotes (TAQ) data originally developed by the New York Stock Exchange is the academic standard for this type of data and has been used extensively in research [22]. This data includes identification information by market fragment (e.g., NYSE, NASDAQ, ARCA, BATS). This type of data contains considerably more information than Level 1 data, but at about a few GB per day it is also quite convenient for analysis and is used in our current work. A sample using TAQ data to describe the Flash Crash event on May 6, 2010 is shown in Figure 1.

2.2.3 Level 3 – Transactions, BBO Quotes, and Limit Order Book (LOB) Information

Level 2 data only shows the inside Best Bid/Offer (BBO) prices in the LOB. There are many limit orders in the LOB as well as prices away from the BBO. With decimal penny pricing replacing the old 1/8 and 1/16, these have become more important. With ever faster electronic market access, they can change very rapidly, in milli- to micro-seconds, so LOB data is usually a snapshot in time, showing prices, and total sizes at each price. In some data, the breakdown of the many orders comprising the total size at a price is also included. Figure 2, which is from the September 2010 SEC/CFTC report, uses color codes to show price levels in snapshots of the time varying "depth of the book."

2.2.4 Level 4 – Transactions + Quotes + LOB + Order Flow

Orders get into the LOB as messages to market systems, and are removed as cancellation messages. This traffic volume can become extremely high. The ever-faster activities of High Frequency Traders (HFTs) are the primary source of this message volume. Algorithmic trading and conventional electronic orders (e.g., retail) comprise the rest. It used to require a phone call to modify, replace or cancel an order in the LOB, a

[1] More information about NERSC is available at `http://www.nersc.gov/`.

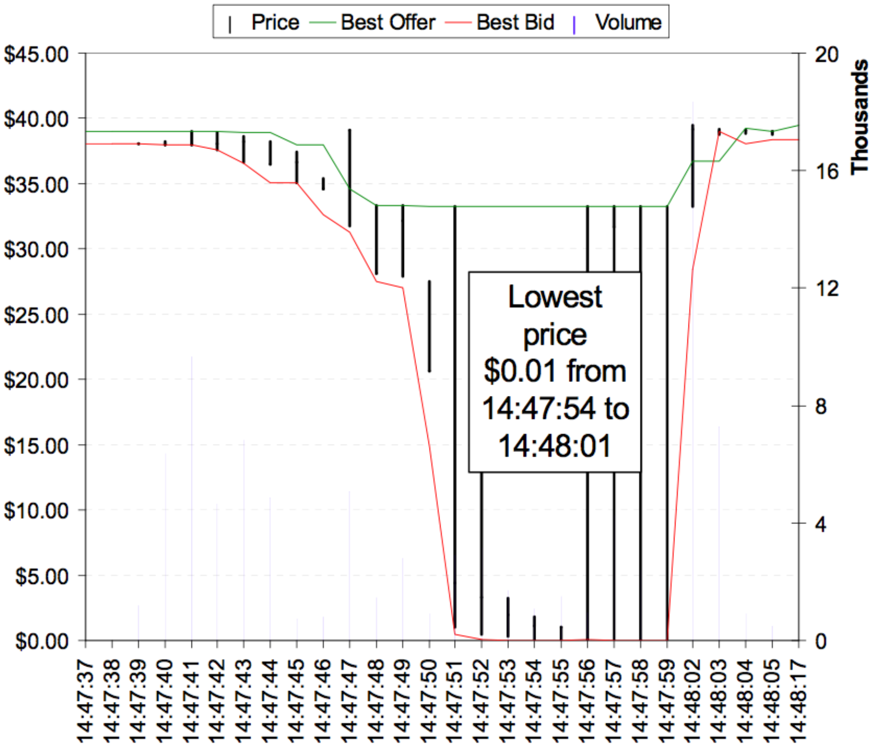


Figure 1: Using Level 2 data shows the bizarre behavior in the trades and quotes for Accenture (ACN), one of the stocks most extremely impacted during the Flash Crash. (Source: May 18 CFTC/SEC report, pg 35)

process that took minutes to complete. It is now measured in units of milli- and micro-seconds. This traffic can at times exceed the capacity of market systems, and result in an unintentional Denial of Service attack.

Nanex, a small high-end market data firm [21] has published an ongoing series of interesting observations at this level of data. A sample is shown in Figure 3.

2.2.5 Level 5 – Identifying information

All of the data discussed so far has been anonymous. A key role for regulators is enforcement, which requires identifying information. The Large Trader Rule and other Legal Entity Identification requirements are designed to accommodate such use. This type of data contains highly sensitive identifying information, and thus engenders strong concerns about privacy.

It is estimated that there are up to 100 market "fragments" for trading of stocks alone. The best known venues, the NYSE and NASDAQ have seen their market shares for their securities drop from more than 90% to the 20% range. That is only for stocks. Futures, options and ETFs introduce further complications, couplings, and cross market issues.

2.2.6 Level 6 – Systemic Engineering Data

None of the data includes any engineering or systems data describing the queues, delays, and traffic that could adversely affect the performance of market systems due to accidental Denial of Service anomalies

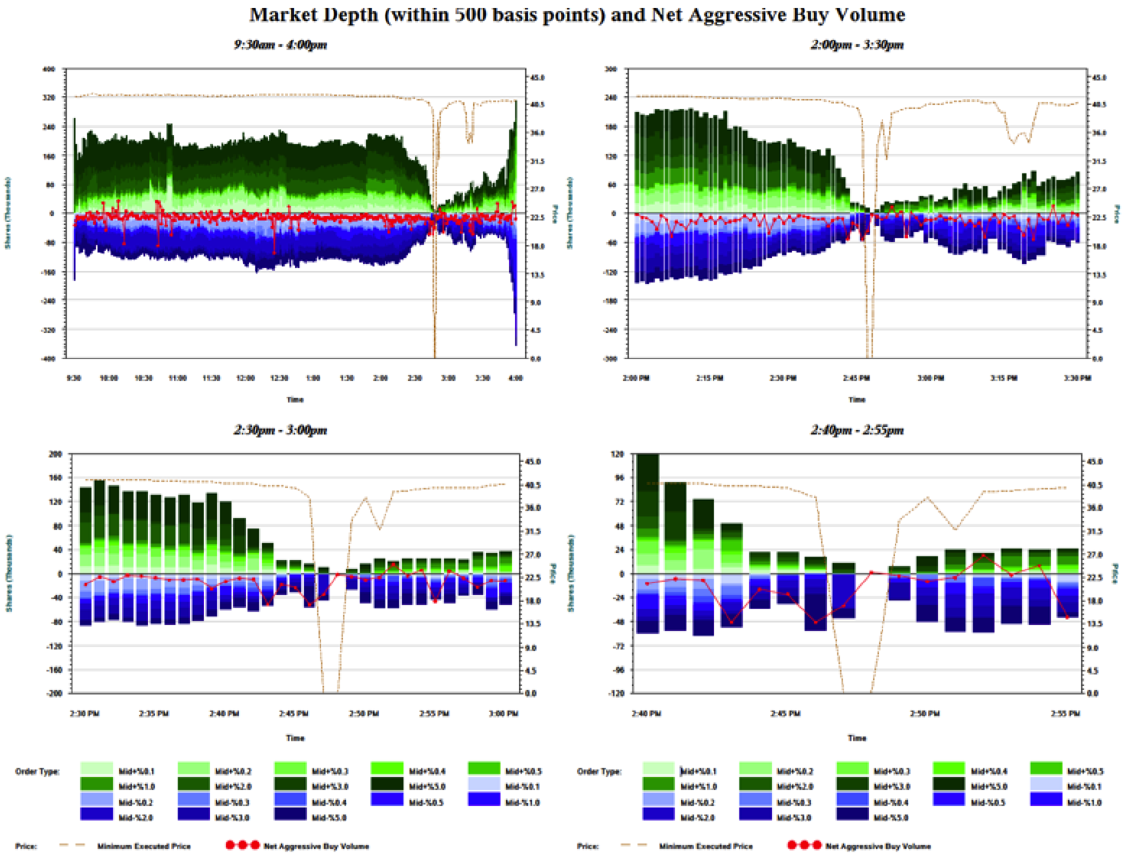


Figure 2: Level 3 data charts showing the evaporation of liquidity from both sides of the the LOB for Accenture (ACN) during the Flash Crash, viewed on four time scales. (Source: Sept 29 SEC/CFTC report, pg 88)

and result in additional undesirable events, such as "mini flash crashes," both up and down, which in fact are still observed.

2.3 Market Indicators for Early Warning of Anomalies

Numerous indicators have been devised to measure aspects of the financial market. Some of these, such as the S&P 500, DJIA and Chicago Board Options Exchange Market Volatility Index (VIX), are used daily in the evening news, while others are only known to the academics and traders interested in specific market activities. In our attempt to detect anomalous activities, we start by examining a few indicators that are known to provide early warning signals for the Flash Crash of May 6, 2010. They are the Volume-Synchronized Probability of Informed Trading (VPIN) [10] and the Herfindahl-Hirschman Index (HHI) [12, 14]. Because these indicators revealed unusual behavior on May 6, 2010, they might be useful in detecting other unusual activities. Both these indicators require substantially more computations than many of the well-known indicators. Because of the computational demands, computing indicators like VPIN and HHI in real-time will require HPC resources. Next, we briefly describe these two indicators.

VPIN measures the balance between buy and sell activities [10]. An earlier version of this indicator is called Probability of Informed Trading (PIN) [9]. The key change in VPIN is to use bins with the same

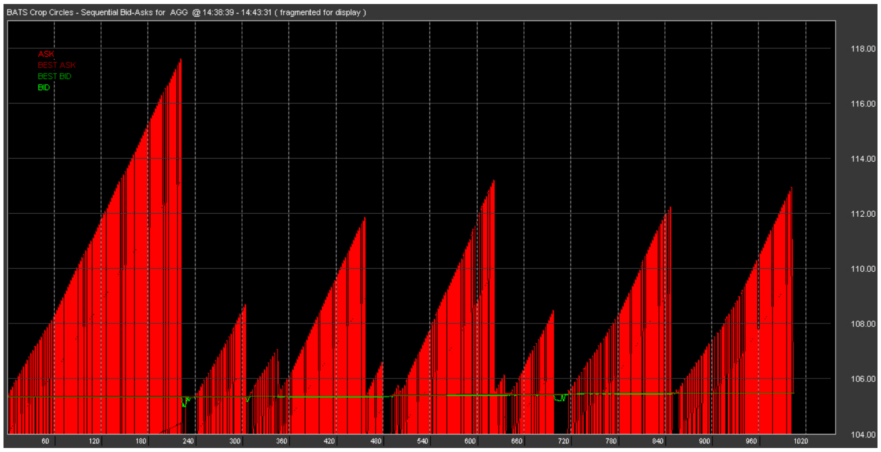


Figure 3: Using Level 4 data, this Nanex "bandsaw" example shows 5 minutes of order flow messages, which modify the LOB at ever faster rates. This activity has been done, albeit at slower speeds, for a very long time, but all computer and communications systems have capacity limits. In extrema, order flow traffic can cause disruptions. (Source: http://www.nanex.net/20100506/FlashCrashAnalysis_Part4-1.html)

trading volume instead of bins with the same time span. The VPIN authors refer to this as measuring the buy-sell imbalance in volume-time instead of clock-time. Furthermore, instead of using the relative imbalance value directly, which can be different for different commodities, the authors normalized them using the function Φ that defines the Cumulative Distribution Function of a normal distribution. Because of this normalization, a single threshold, $T = 0.9$, can be used for many different stocks. With suitable parameters, the authors have shown that the VPIN reaches 0.9 more than an hour before the Flash Crash on May 6, 2010. This is the strongest early warning signal known to us at this time. To a large extent, our computation of VPIN replicates the one used by Easley et al. [9]. In fact, the developers of VPIN have shared a Python implementation of their program with us and our C++ implementation reproduces exactly the same values on the same input values. A key difference is that we compute VPIN values of individual stocks while the earlier work computes VPIN values on SP500 futures.

Another indicator producing a clear early warning signal for the Flash Crash of 2010 was a market fragmentation measure based on HHI [20]. The particular version used in this work is called the Volume Herfindahl Index, but many other variations exist in the literature. Because we will not use any other type of HHI, we simply refer to it as HHI in later discussions. During a given time window, say five minutes, the fractions of trade volumes executed by different stock exchanges can be computed. HHI is the sum of squares of these fractions [12, 14]. Variations of HHI are widely used to measure the concentration of industrial production or other economic power [8, 29]. In this case, the particular version of HHI measures how concentrated the exchange operations are. An HHI value of 1 indicates that all trades are executed at a single exchange, which is easiest for buyers to be matched with sellers. When the value of HHI is smaller than 1, some fractions of trades are executed at different exchanges and potentially at unequal prices. In other words, buyers and sellers at different exchanges may be treated differently. This phenomenon is generally referred to as market fragmentation and is considered as a source of market instability.

In [20], Madhavan computes a single HHI value for an entire day. In an attempt to use HHI as an early warning indicator, Madhavan suggested that we break each day into small intervals. In our work here, we choose to use 5-minute bins. Furthermore, to detect "abnormal" values, we define a reference window covering twelve bins that covers the hour preceding the current bin. We use the bins in the reference window to compute a mean and a standard deviation. We declare an HHI to be "abnormal" if it is more than x times the standard deviation away from the mean. Note that the x value of 1.645 is equivalent to the choice of 0.9 as the threshold for VPIN.

In this paper, we examine how HPC can accelerate the computation of the indicators. We plan to evaluate the effectiveness of the indicators in the future.

2.4 Data Management

In order to perform the above mentioned computations effectively, the required data must be in the appropriate format. For example, the widely available TAQ (Trades And Quotes) data [22] is available typically on a CD or DVD with extraction program that runs only on MS Windows platforms. Since most of the HPC systems run Linux operating systems, TAQ data requires a transformation step before it can be used on HPC platforms. Many other collections of data have a similar limitation.

Even if the computing platform is based on MS Windows, the data extraction program, such as the one provided with the TAQ data distribution, produces Comma-Separated Values (CSV), which is an ASCII representation of the values. This representation typically requires more bytes than the corresponding binary representation and requires significantly more time to read into memory. Furthermore, because the data records in ASCII format typically require different numbers of bytes, it is more difficult to skip unwanted bytes to directly extract a specific data record without also reading the records preceding it in the data file. Other data distributions, such as the one from Nanex, do not require the intermediate step of converting data to ASCII format. However, many of them still require the user to go through each data record, without obvious means for skipping unwanted records.

One way to provide high-performance data access is to store the data records in a commercial database management system (DBMS). Some DBMS have extensive support for operations on financial data series. However, to achieve higher performance and to have more control over the analysis operations, we have taken the approach used by many scientific applications – using a high-level data format library, the Hierarchical Data Format version 5, or HDF5 [11].

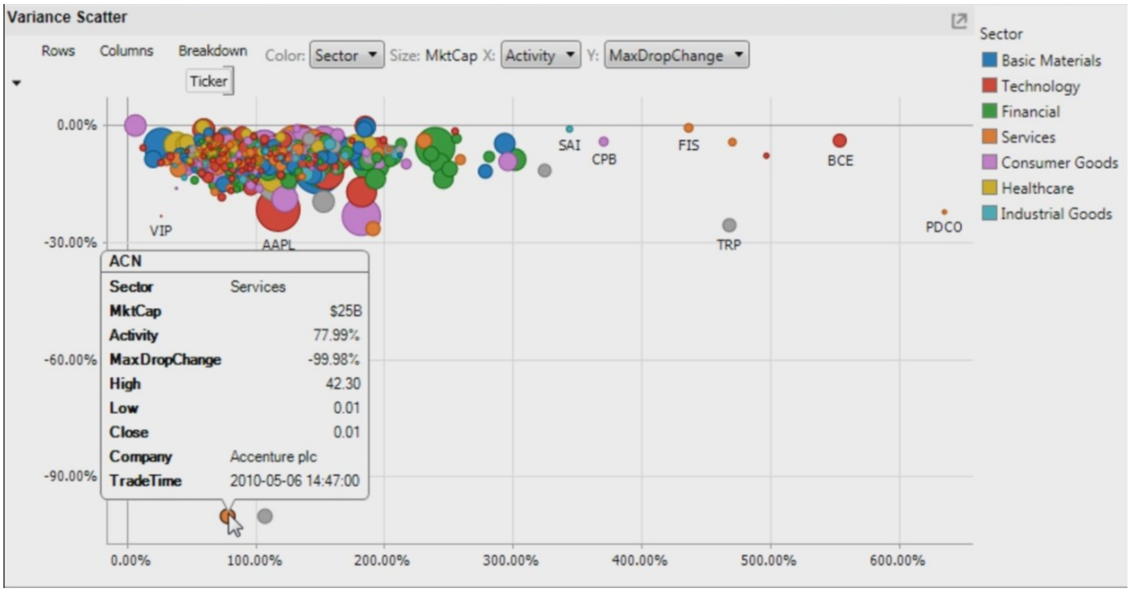
2.5 Data Exploration, Visualization, and Simulation

Data exploration, visualization, and simulation are key concepts in scientific HPC. A detailed discussion of these topics in a fast financial context is beyond the scope of this paper. However, we want to include an example that dramatically illustrates the truly strange nature of the Flash Crash, and some of the remaining open questions.

Figure 4 shows a still frame from a movie produced by the visualization firm Panopticon. The movie, even though very instructive, is backward looking – showing what happened. A key question for market regulators is if we can anticipate future flashes and other anomalies before they occur. Stopping trading through the use of "circuit breakers" is current practice. Alternatively, one could slow down trading using a "yellow light" mechanism that would allow HFT programs and other algorithmic market actors to back off gracefully.

3 Case Study

The goal of this case study is to evaluate how high-performance computing can support financial data analysis and, in particular, the development and implementation of early warning systems for detection and



Color: Sector Size: MktCap X: Activity Y: MaxDropChange

ACN

Sector	Services
MktCap	$25B
Activity	77.99%
MaxDropChange	-99.98%
High	42.30
Low	0.01
Close	0.01
Company	Accenture plc
TradeTime	2010-05-06 14:47:00

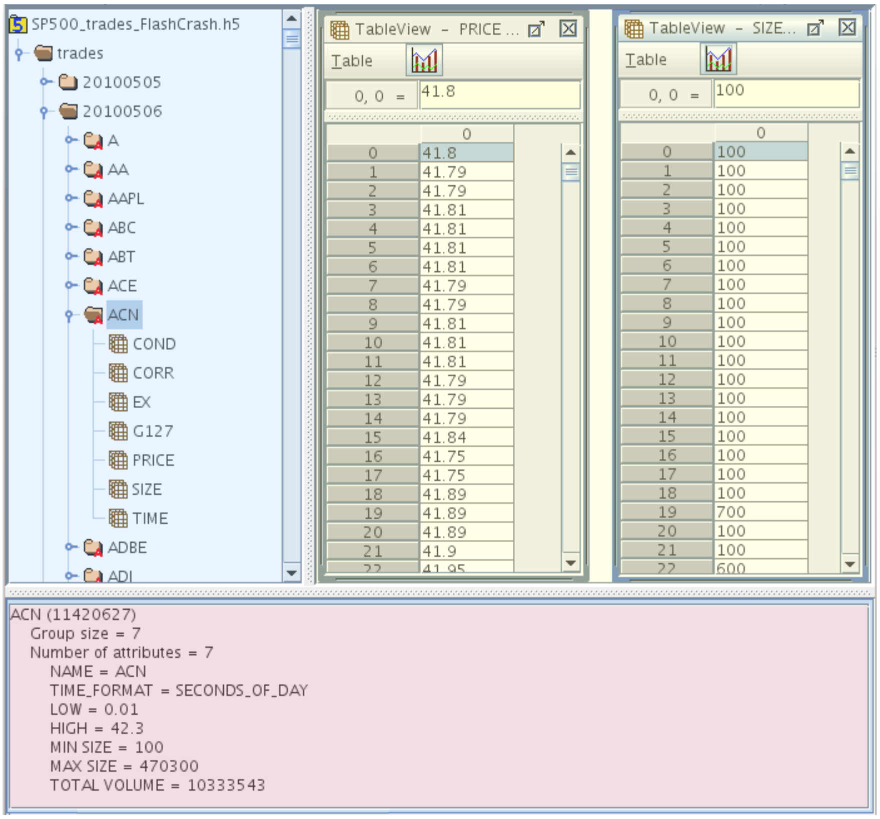
Figure 4: A view showing the price changes and volumes for S&P 500 stocks during the Flash Crash. Sectors are show by the color coded circles, market capitalization by the size of the circle, price change is on the vertical axis, and volume on the horizontal. ACN, which is also shown in Figure 1 and 2, is selected on the lower left . (Source: Panopticon)

analysis of market anomalies. Development and evaluation of reliable indicators for market anomalies requires thorough analysis of the effectiveness of such indicators on large amounts historic data. We need to be able to: i) store and process large amounts of data, ii) efficiently compute market indicators, and iii) quickly extract and analysis portions of data during which abnormal market behavior is indicated.

3.1 File Format

Enabling efficient analysis of large amounts of data fundamentally relies on effective data organization and storage to optimize I/O performance and allow algorithms to quickly locate data records of interest. Financial data is commonly stored in simple ASCII and binary formats, like TAQ data, or proprietary binary data formats. Simple ASCII and binary formats are easily accessible but inefficient with respect to storage requirements, do not support efficient search operations, are not portable (binary data only), and are inefficient for parallel I/O. Proprietary data formats often suffer from similar problems, are often opaque to the user, and require the use of commercial, black-box libraries for file access.

Over the course of the last decades, the HPC community has developed advanced, open scientific data formats that address many of the data challenges the financial community is facing. In this work we adopt HDF5 [11] – a state-of-the-art, open, scientific data format – for storing financial data. Figure 5 illustrates the HDF5 data-layout we are using for storing TAQ data. We organize the data into groups based on the data type ($/trades$, $/quotes$), date, and stock symbol. Each complete group (e.g. $trades/20100506/ACN$) then contains a set of 1D HDF5 datasets of varying types (e.g., $PRICE$ stored in floating-point format, or $SIZE$ stored as integers). Additional information about the data, like the time format, and simple statistics, such as minimum price or total volume, are stored as HDF5 attributes associated with the corresponding datasets and groups. SZIP compression [32] can further significantly reduce file size, by a factor of 5-7 in



0, 0 = 41.8

	0
0	41.8
1	41.79
2	41.79
3	41.81
4	41.81
5	41.81
6	41.81
7	41.79
8	41.79
9	41.81
10	41.81
11	41.81
12	41.79
13	41.79
14	41.79
15	41.84
16	41.75
17	41.75
18	41.89
19	41.89
20	41.89
21	41.9
22	41.95

0, 0 = 100

	0
0	100
1	100
2	100
3	100
4	100
5	100
6	100
7	100
8	100
9	100
10	100
11	100
12	100
13	100
14	100
15	100
16	100
17	100
18	100
19	700
20	100
21	100
22	600

```
ACN (11420627)
   Group size = 7
   Number of attributes = 7
      NAME = ACN
      TIME_FORMAT = SECONDS_OF_DAY
      LOW = 0.01
      HIGH = 42.3
      MIN SIZE = 100
      MAX SIZE = 470300
      TOTAL VOLUME = 10333543
```

Figure 5: HDF5 data-layout used for storing trades data showing: i) the organization of the data via HDF5 groups and datasets (blue), ii) example meta-data for ACN, 05/06/2010 (red), and iii) spreadsheet-view of PRICE and SIZE data array for ACN, 05/06/2010 (yellow).

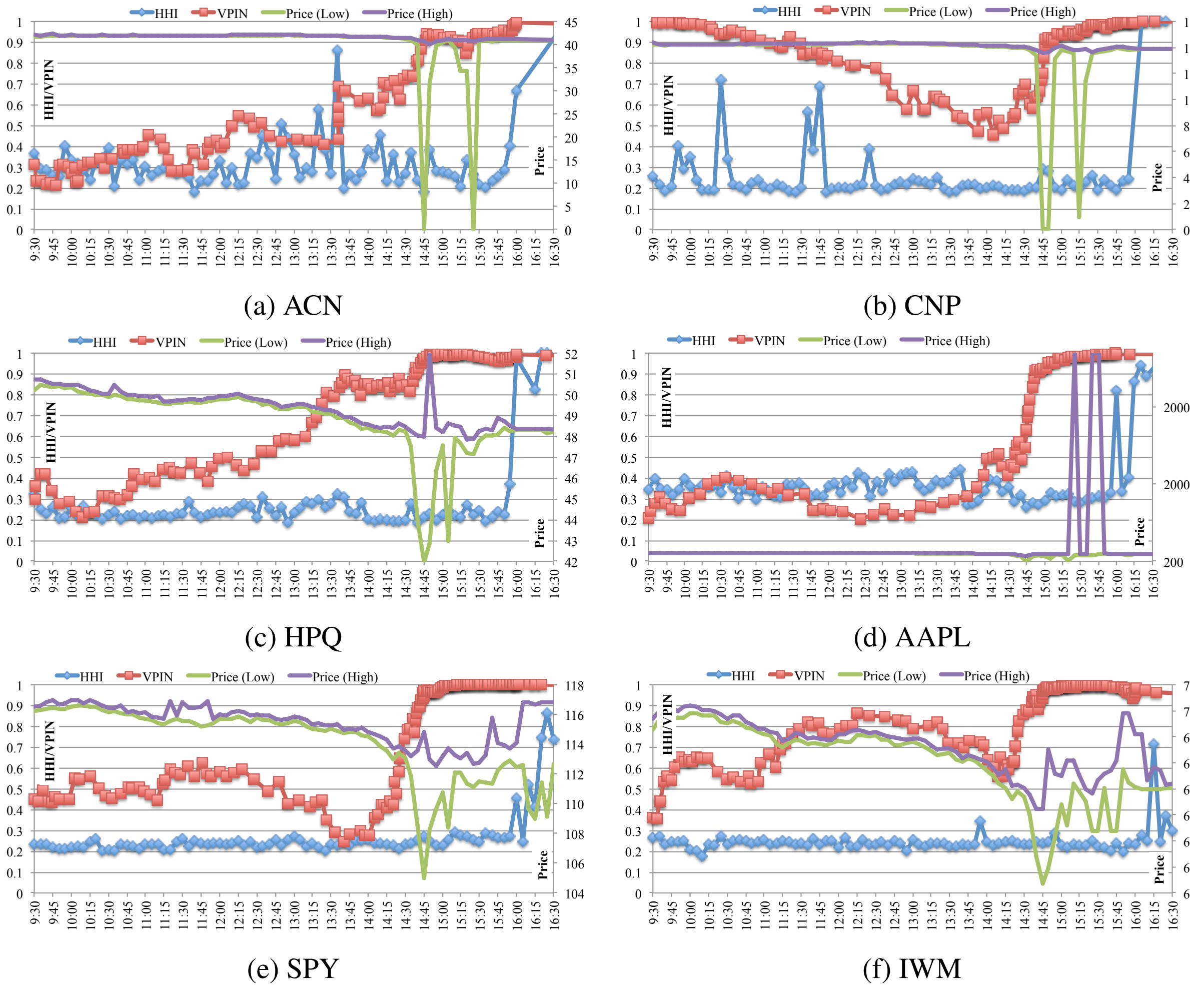
	CSV	**CSV (zip)**	**HDF5**	**HDF5 (SZIP)**	**Index**
Trades	2,769	215	1,326	472	1,803
Quotes	38,566	3,058	28,844	5,377	24,784

Table 1: File sizes in mega-byte (MB) for example TAQ data using different file formats. The datasets contain three days worth of trades and quotes for S&P 500 symbols.

the case of TAQ data, while enabling fast decompression and, hence, data access. In contrast to manually compressed CSV files, SZIP compression is integrated transparently with HDF5: the user does not need to explicitly decompress the data first but can access the compressed data directly through the HDF5 API. Table 1 compares the storage requirements of an example TAQ dataset in different formats.

Using HDF5 for storing financial data has many advantages. HDF5 is portable, easy to use, efficient with respect to storage and I/O performance, supports compression and parallel I/O, and provides additional tools for browsing, validation and profiling (e.g, HDFView). HDF5 is optimized for large data files and enables us to store months to years worth of financial data in a single file while maintaining easy data access.

The ability to group data into meaningful subgroups within a file improves the data organization, eases data access, and reduces storage cost for highly repetitive data fields, like date and stock symbol. Using groups also allows us to store related but heterogeneous financial data types, such as trades and quotes



(a) ACN

(b) CNP

(c) HPQ

(d) AAPL

(e) SPY

(f) IWM

Figure 6: HHI (blue) and VPIN (red) values on May 6, 2010. The minimum and maximum prices shown are computed in each 5-minute bin. Each indicator shows extreme values before the irregular price changes.

data, in the same file. The ability to augment individual HDF5 groups and datasets with meta-data allows us to store additional information about the data, such as daily statistics, in an easily accessible manner. The organization into groups of multiple HDF5 datasets, one per variable, makes it convenient to access selected attributes of a trade or a quote. For example, the computations of both VPIN and volume HHI use information on trades; however, neither of them require all eight attributes. We only need three attributes to compute either VPIN or HHI: time, price and volume for VPIN, and time, volume and exchange for HHI. However, dividing the data into a large number of subgroups, each containing only $\approx 10,000 - 200,000$ records in the case of TAQ, may reduce the effectiveness of compression and may require a large number of disjoint data access operations, which in turn reduces overall I/O performance. Depending on the application, one may be able to achieve higher performance, by grouping data into larger subsets by grouping records by week or month rather than by day.

One benefit of using the above organization is that it enables convenient use of advanced HPC index and search software. In our case, we used the FastQuery software to index and search the data [7]. FastQuery exploits the searching capabilities of a set of bitmap indexes implemented in FastBit [31] to a number of popular scientific data formats including HDF5. In the past, this indexing technique has been successfully used to search over historical data to confirm the observations made in real-time data activities [27], to perform forensic cybersecurity analysis [2,26], and to enable use of HPC resources for large-scale scientific data analysis [24]. In the application area of financial data analysis, we believe that the bitmap indexing technique can fulfill a similar role of searching historical data to efficiently compute early warning indicators. This idea is the subject of the next section.

3.2 Computing Market Indicators

After organizing the data in an effective way, the next question to answer is whether HPC resources can effectively compute market indicators. We use the computations of VPIN and HHI as examples, and treat the computation of VPIN and HHI on each stock or fund as a separate computational task. Because these tasks don't require any coordination among them, they can achieve good speedup as we show later. A key limitation to achieving perfect speedup is that the amount of work in each computational task is very different, resulting in load imbalance. Therefore, a dynamic task scheduling method is needed to balance the amount of work assigned to each process. In this work, we use a manager-worker approach to distribute the tasks. In this section, we also present some evidence that HHI and VPIN produce strong signals before and during the Flash Crash of 2010.

We start our discussion on the computation of VPIN and HHI by describing the data used. We use two different sets of trades data. The first set of trades covers the time period of April and May, 2010, containing 45 trading days. This set contains all trades of SP500 stocks. The total number of records is about 640 million and the total size is about 25 GB as CSV files and 4.4 GB as HDF5 files.

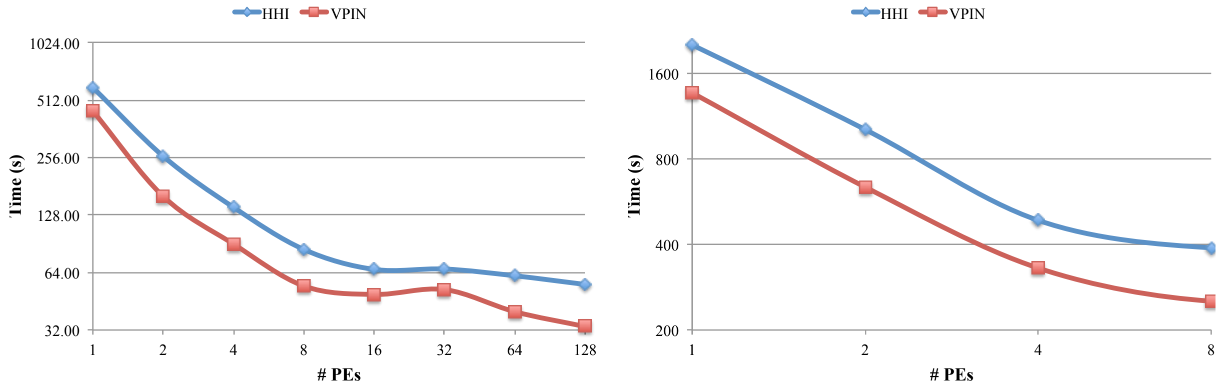
The second set contains trades of 25 ETFs with the largest trading volumes. The time period varies from 3 years (2008, 2009, and 2010) to 10 years (2001 – 2010). The total number of records is about 2.7 billion, and the size is 108 GB in CSV and 17 GB in HDF5. Clearly, there is a size advantage for using HDF5 files.

It is faster to use data in HDF5 files for computation as well. For example, on a subset of data from May 2010, using HDF5 it only took 0.4 seconds to compute VPIN for Accenture (ACN) stock. However, it took 142 seconds using the corresponding CSV files. Using HDF5 files speeds up the VPIN computation by a factor 355. The key difference between using HDF5 and CSV is that using HDF5 files, combined with efficient indexing, one can quickly locate the desired data records, while using CSV files, one has to read through each data record to locate the desired records about ACN.

We compute VPIN and HHI for each stock or fund separately, in order to raise the "yellow flag" on each of them independently. We realize that this is not exactly how the original authors of VPIN and HHI intended to use them [10,20]. However, as we show in Figure 6, there is strong evidence that VPIN and HHI can indeed provide early warning for the Flash Crash of May 6, 2010.

In Figure 6 we show the values of VPIN and HHI for four individual stocks and two ETFs. The stocks are Accenture (ACN), CenterPoint Energy (CNP), Hewlett-Packard (HPQ), and Apple (AAPL). The first two are the well known examples where the prices dropped to one penny per share during the Flash Crash, HPQ is one of Dow-Jones stocks significantly affected by the Flash Crash, and AAPL stock has the unusual behavior of reaching to $100,000 per share during the same time period. The ETFs shown are SPY and IWM, these are the closest relatives of the futures used by the original authors of VPIN [10].

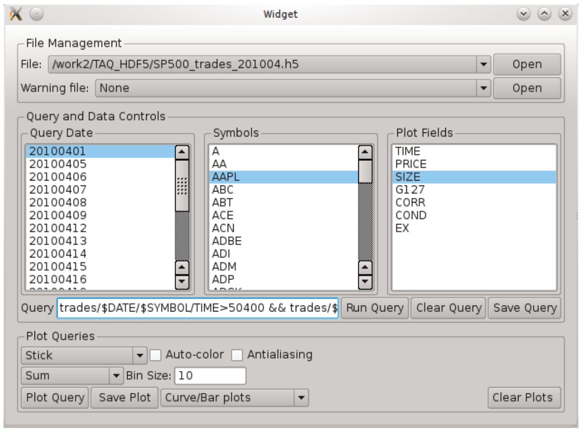
The data shown in Figure 6 is from May 6, 2010, between 9:30 and 16:30. For each 5-minute time interval used to compute HHI, we also compute the minimum price and maximum price during the same time interval. In Figure 6, we see that the minimum price drops to one penny in a number of bins, the earliest of which is around 14:45. Three of the four stocks shown exhibit significant price drops. There are other



(a) SP500 stocks

(b) Top 25 ETF

Figure 7: Time (seconds) needed to compute HHI and VPIN using different number of processing elements (PEs). The computation is broken up into 500 independent tasks for SP500 data and 25 tasks for the 25 ETFs. The maximum speedup achieved is about 11 for computing HHI and 13 for computing VPIN on SP500 data and about 5 for computing both HHI and VPIN on the 25 ETFs.



File Management
File: /work2/TAQ_HDF5/SP500_trades_201004.h5 Open
Warning file: None Open

Query and Data Controls

Query Date: 20100401, 20100405, 20100406, 20100407, 20100408, 20100409, 20100412, 20100413, 20100414, 20100415, 20100416

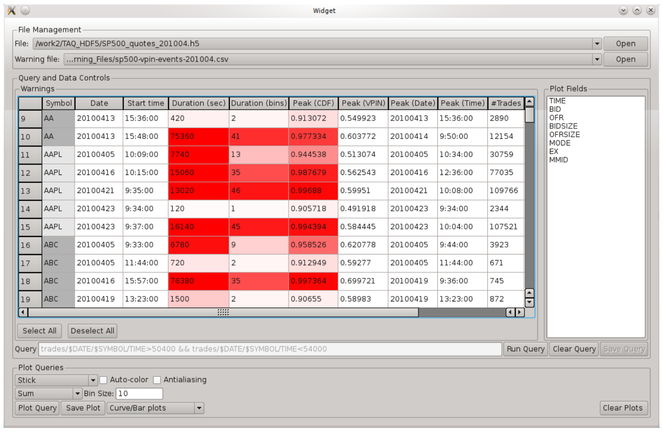
Symbols: A, AA, AAPL, ABC, ABT, ACE, ACN, ADBE, ADI, ADM, ADP

Plot Fields: TIME, PRICE, SIZE, G127, CORR, COND, EX

Query: trades/$DATE/$SYMBOL/TIME>50400 && trades/$

Plot Queries
Stick ☐ Auto-color ☐ Antialiasing
Sum Bin Size: 10

Figure 8: User interface for defining symbolic trades/quotes queries.



File Management
File: /work2/TAQ_HDF5/SP500_quotes_201004.h5 Open
Warning file: ...rning_Files/sp500-vpin-events-201004.csv Open

Query and Data Controls

Warnings

	Symbol	Date	Start time	Duration (sec)	Duration (bins)	Peak (CDF)	Peak (VPIN)	Peak (Date)	Peak (Time)	#Trades
9	AA	20100413	15:36:00	420	2	0.913072	0.549923	20100413	15:36:00	2890
10	AA	20100413	15:48:00	75360	41	0.977334	0.603772	20100414	9:50:00	12154
11	AAPL	20100405	10:09:00	7740	13	0.944538	0.513074	20100405	10:34:00	30759
12	AAPL	20100416	10:15:00	15060	35	0.987679	0.562543	20100416	12:36:00	77035
13	AAPL	20100421	9:35:00	13020	46	0.99688	0.59951	20100421	10:08:00	109766
14	AAPL	20100423	9:34:00	120	1	0.905718	0.491918	20100423	9:34:00	2344
15	AAPL	20100423	9:37:00	16140	45	0.994394	0.584445	20100423	10:04:00	107521
16	ABC	20100405	9:33:00	6780	9	0.958526	0.620778	20100405	9:44:00	3923
17	ABC	20100405	11:44:00	720	2	0.912949	0.59277	20100405	11:44:00	671
18	ABC	20100416	15:57:00	76380	35	0.997364	0.699721	20100419	9:36:00	745
19	ABC	20100419	13:23:00	1500	2	0.90655	0.58983	20100419	13:23:00	872

Plot Fields: TIME, BID, OFR, BIDSIZE, OFRSIZE, MODE, EX, MMID

Query: trades/$DATE/$SYMBOL/TIME>50400 && trades/$DATE/$SYMBOL/TIME<54000

Plot Queries
Stick ☐ Auto-color ☐ Antialiasing
Sum Bin Size: 10

Figure 9: User interface for evaluating market indicator-based warnings.

noticeable price variations later in the trading day as shown in Figure 6. Not all the stock prices fell during the Flash Crash, in Figure 6 we see a small jump for HPQ and two sets of tremendous jumps for AAPL.

In Figure 6, the values of VPIN became quite high before the Flash Crash at 14:45. In the case of ACN, there was a sharp rise for both HHI and VPIN at 13:35[2]. This was about 70 minutes before the Flash Crash, slightly before the time frame examined in detail in the official report on the Flash Crash, which suggests that both VPIN and HHI might have detected some signals missed by the investigators.

For other stocks, either VPIN or HHI showed similar early warning before the Flash Crash. For example, for CNP, the VPIN values were very high earlier in the day, for HPQ, the VPIN values also reached a high level around 13:45 about an hour before the Flash Crash, for AAPL, the VPIN value reaches a high level at

[2] The particularly sharp rise in VPIN and HHI is directly linked to a unusually large trade at 13:36:07. The volume of this single trade is 470,300 shares, which is almost 10% of the average daily volume for ACN. The impact of such a large trade on VPIN and HHI needs to be further examined.

14:45 about half an hour before the unusual event at 15:15. Combined with the evidences provided by other authors [10, 20], we believe that VPIN and HHI are strong candidates for providing early warning signals of unusual activities.

Figure 7 shows how the time needed to compute HHI and VPIN varies with the number of processors used. The system used has 24 CPU cores on a compute node and each node has two network interface to other compute nodes and file systems[3]. In Figure 7(a), we show the time to compute HHI and VPIN on SP500 stocks. In this case, the task of computing HHI or VPIN on each stock is given to a process that uses one CPU core on a node. These tasks are independent from each other in terms of computation, however, they need to access the same file system in order to read the necessary input data and write the final output. In Figure 7(b), we show the time to compute HHI and VPIN on the 25 ETFs. Again, we distributed the task of computing HHI or VPIN for each ETF to a process. since there are only 25 tasks, we used a maximum of 8 processing elements.

Overall, as the number of processes increase, the time needed goes down. However, due to load imbalance among the tasks and performance variance of the file system, time does not vary smoothly. For example, in Figure 7(a), as the number of PEs vary from 16 to 32, the total execution time appears to remain the same or slightly increase.

In Figure 7(a), using 16 processes, the speed up over using one process is about 9x when computing both HHI and VPIN. Using 128 processes, we achieve a speedup of 11x when computing HHI, and 13x when computing VPIN. In Figure 7(b), the maximum speedup of 5x is achieved with eight processes. In both cases, we see the task of computing HHI and VPIN are indeed independent tasks and can be parallelized effectively. As we extend our work on to more stocks and ETFs, we see more opportunity for parallel computation and more efficient uses of HPC resources.

3.3 Query-driven Analysis and Data Exploration

Typical scientific applications require single (or few) queries to be evaluated on extremely large data. In contrast, screening of financial data requires evaluation of a large number of independent data queries (one query per $date/symbol$ combination). Similarly, validation of market indicators fundamentally relies on the ability to quickly locate and extract data associated with large numbers of indicated warning periods, e.g, for HHI we find 298,956 potential warnings for S&P 500 stocks during April 2010.

To allow analysts to quickly define large sets of queries, we extend the FastQuery query language using symbolic queries. A symbolic query is a compact representation of a large number of queries using reserved keywords (here $\$DATE$ and $\$SYMBOL$) to represent data categories. The user can then select, from simple lists, the specific dates and symbols for which a symbolic query should be executed (Figure 8). The symbolic query is then expanded into $\#dates * \#symbols$ queries on behalf of the user. We use standard spreadsheet and statistics plots (Figure 10) for validation of queries and market-indicator warnings.

In the case of market indicators, large sets of warning events are created automatically by the screening process. Each warning has an associated date, symbol, and time period. To extract the data associated with warning events, we automatically translate warnings to queries of the form:

$$\begin{array}{l} TYPE/\$DATE/\$SYMBOL/TIME >= start\&\& \\ TYPE/\$DATE/\$SYMBOL/TIME <= stop, \end{array} \tag{1}$$

where $TYPE$ identifies the trades/quotes group. We use a spreadsheet representation to allow the analyst to quickly browse and select warnings of interest (Figure 9). The duration and peak warning value (e.g.,

[3] More information about the particular computer can be found at `http://www.nersc.gov/systems/hopper-cray-xe6/`.

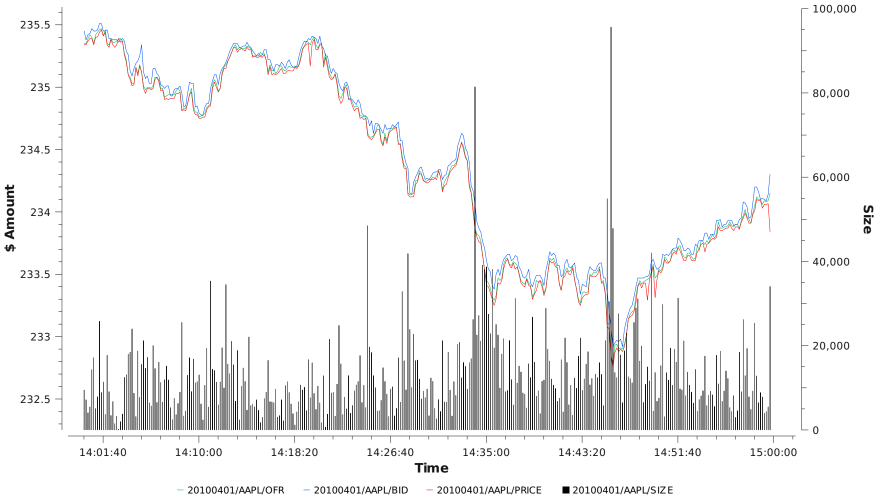


Figure 10: Plot created using our query tool showing the best OFR (green), BID (blue), PRICE (red), and total traded volume (SIZE)(black).

peak $CDF(VPIN)$ value) are color-coded to ease identification of long and high-risk warnings. Once a warning query has been executed, additional information about the corresponding warning is displayed in the table, such as the number of trades and quotes executed during the warning period. To help reduce the analysis of false alarms, we plan in the future to extend this concept with automatic methods for defining whether an anomaly occurred during a warning period.

In the following, we study the parallel performance of our query system, using the Hopper [2] system at NERSC. In the scaling experiment, we evaluate 8000 independent queries as defined in Eq. 1 on S&P 500 quotes data for April 2010. Each date/symbol group contains on average approximately 274,151 records, and 47 out of 8838 of these date/symbol groups contain between 1.5 - 3.9 million records (Figure 11). This constitutes 74.5 GB of uncompressed HDF5 data (13.4 GB as HDF5 with SZIP compression and 100.4 GB as uncompressed csv) and 28.3 GB for all FastBit indices. To evaluate the expected performance on more larger amounts of financial data, we replicated each HDF5 dataset 10 times, effectively increasing the number of quotes executed per date and symbol by a factor of 10. The replication also emulates storing ten month's worth of quotes data grouped into periods of ten trading days. The replicated dataset constitutes 744.7 GB of uncompressed HDF5 data and 210.2 GB for all FastBit indices. Both quotes HDF5 files were created using default HDF5 settings and are stored on Hopper's Lustre file system using a striping of 24 and a stripe-size of 1MB. We use a controller-worker-type setup implemented in MPI to parallelize the query process. The controller schedules the queries in batches of 10 queries as workers become available. Once all queries are completed, the controller acquires all hit-counts from the workers, while the file offsets (results) are stored on the workers to allow for efficient parallel analysis of the data associated with the queries. We repeated each experiment ten times and report the average wall-clock time elapsed to evaluate all queries, including all communication.

Figure 12 shows the results of this parallel query study. We observe good scalability in all cases and achieve two orders of magnitude speedup compared to the serial case. Using the CSV data, it takes approximately 3.5 hours in serial to evaluate all 8000 queries, even though we can compute all queries in a single pass, exploiting the fact that data records are sorted by date, symbol and time in the CSV file(s). Using HDF5 and indexing, we are able to perform the same analysis in parallel in less than 5 seconds on both

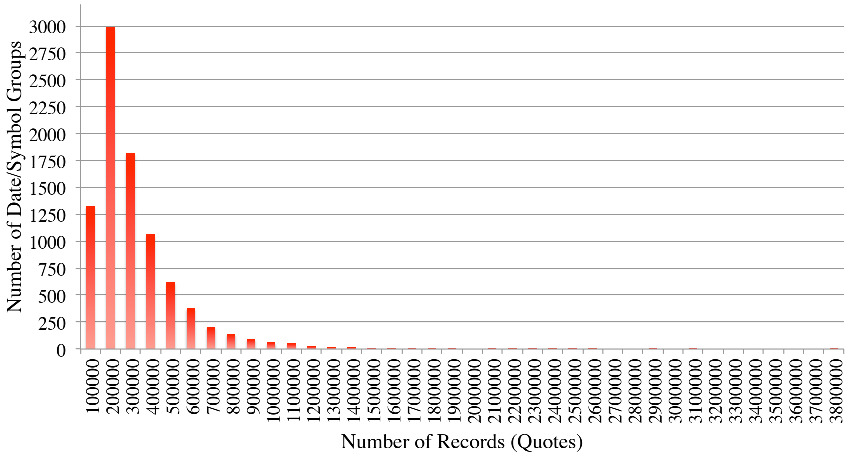


Figure 11: Number of data records (quotes) per date/symbol group for S&P 500 quotes data for April 2010.

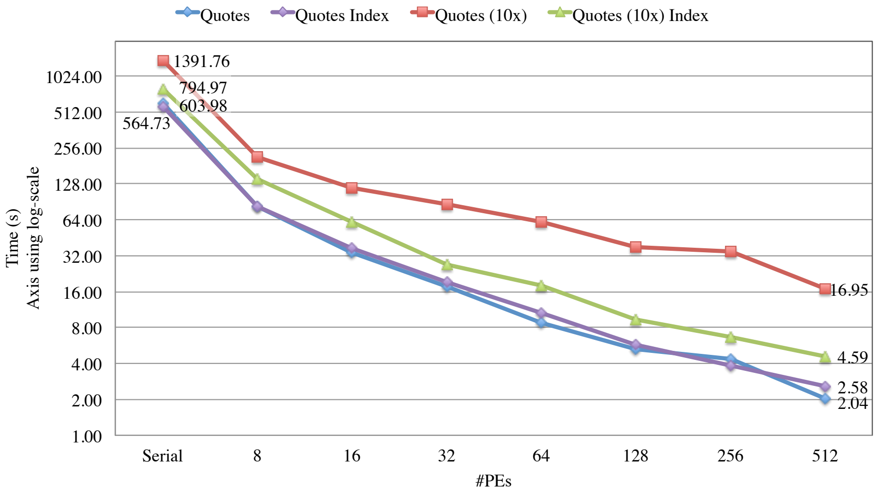


Figure 12: Time (seconds) needed to evaluate 8000 queries using different numbers of processors on i) S&P 500 quotes data for April 2010 (lilac) and with indexing (blue) and ii) the same dataset with 10× replication (green) and with indexing (red). Note the log-scale on the vertical time axis.

datasets, constituting a three orders of magnitude speed-up compared to using CSV. For the regular quotes data we do not observe a significant difference between the version with and without FastBit indexing. This behavior is expected, since the number of records required per query is low. For the dataset with 10× replication, we observe speedups of $\approx 2 - 4$x when using indexing. Interestingly, the time used for evaluating queries on the 10× replicated data does not increase by a factor of 10 compared to the regular quotes data. This is likely due to inefficiencies with respect to the HDF5 write: the default chunk-size may be too large for the small numbers of records per dataset, and/or the filesystem performance, which is sensitive to tunable parameters like the level of striping and stripe size, resulting in a larger I/O overhead for the smaller dataset. This type of variation in performance is well known: it has been shown that tuning of HDF5 and file system parameters can have a significant impact on I/O performance [15].

4 Discussion

Data organization In this work, we demonstrated the benefit of using a more efficient data organization by using the HDF5 file format. For computing market indicators (Section 3.2), we see a speedups of about 355x by using HDF5 compared to using CSV. This speedup results directly from more efficient data access methods. Organizing and storing the financial data in a more efficient way has a tremendous potential of improving the monitoring and reporting of financial markets.

Parallel computing Large-scale analyses of financial data – such as query-based analysis of historical data – require processing of large amounts of computationally independent data, e.g., data for different stocks. The experiments in Sections 3.2 and 3.3 have shown that we can parallelize these types of operations effectively. The combination of efficient data organization and parallel computing enables us to evaluate large amounts of queries in seconds rather than hours. Such results, which stem from a combination of parallel computing, efficient data I/O and index/search, are consistent with other studies in forensic cybersecurity analysis [2, 26] and large-scale scientific data analysis [24], where processing time was reduced from hours or days to minutes or seconds. We are able to evaluate 8000 queries on a 74.5GB dataset in $\approx$2 seconds, an operation that takes more than 3.5 hours in serial using CSV data, a speedup of about 6,300x. The same operation on the larger 744.7GB dataset still requires less than 5 seconds.

HPC for early warning systems Based on our preliminary results, we believe that early warning systems can benefit substantially from HPC research, systems, and tools. Evaluation of the effectiveness of potential early warning indicators requires screening of large amounts of historical data. Ultimately we need to be able to compute market indicators in real time, requiring massively parallel algorithms and high-throughput data networks to analyze data from large numbers of stocks at once. At the same time, the real-time data needs to be stored efficiently for later analysis. For example, to enable regulators to judge the credibility of an indicated alarm, they need to be able to quickly locate and analyze similar events in large amounts of historical data. While our case study was limited in scope, focusing only on TAQ data, the results indicate that HPC methods can facilitate many of the tasks necessary for development, operation, and monitoring of a market monitoring and alarm system. For use scenarios involving larger amounts of financial data, we expect the computational demands to increase significantly making HPC methods indispensable.

Future Work In the current work, we have encountered inconsistent data from different sources. For example, different sources disagree on how many trades of AAPL at $100,000 per share occurred on May 6, 2010. In the TAQ data, there are four records of trades at this price, two at 3:29:30 PM with a total of 895 shares, one at 3:44:51 PM with 695 shares, and one at 3:49:39 PM with 200 shares. In the Nanex data, there are also four trades recorded, though the volumes match those from TAQ, all of the records in Nanex data have the time stamp of 3:29:30 PM. The official SEC/CFTC report about the Flash Crash only mentioned two trades at 3:29:30 PM with a total volume of 895 shares. Clearly, such a discrepancy is a serious issue. Increasing the data quality should be one important goal for improving transparency and efficiency of financial markets.

With respect to data management, we plan to further investigate improvements of the HDF5 data layout, and tuning of HDF5 and filesystem parameters to improve I/O performance and efficiency of data indexing methods.

The current version of VPIN requires all trades to be present to determine the function Φ needed for the final normalization. We plan to develop a variation of VPIN for real-time computations. It may also be fruitful to evaluate whether using volume-time for binning can improve the effectiveness of HHI.

Quantification of the effectiveness of potential early warning indicators fundamentally relies on the ability to judge whether a warning is true or false and whether relevant anomalous behavior is missed by an indicator. We, therefore, plan to develop algorithms to automatically detect anomalies in historical financial data. Evaluation of market indicators and validation of designs and implementations of real-time market monitoring systems will also benefit from realistic high-performance simulations of financial markets.

5 Conclusion

This “early warning” line of inquiry begins to address a key question regarding the role of high-performance computing in finance from a federal perspective: Is real-time high frequency monitoring needed? The SEC/CFTC has announced their intention to direct many billions from the financial industry to this effort, which has been criticized by others as unnecessary overkill.

We and our collaborators have come to believe that it is not overkill. Current post Flash Crash regulatory approaches are based on “circuit breakers,” which suspend trading when price or volatility triggers set them off. These are very “blunt instruments” that do not allow the market to self-correct and stabilize, and they can easily make a bad situation worse. Our tests showed that VPIN, HHI and similar indicators could provide early warning signals for a more gradual “slow down, rather than stop” replacement for on/off circuit breakers. Our HFT and academic collaborators hold this opinion strongly as well.

This work explores a number of pressing issues in implementing such an “early warning” system, such as the need for sufficient computing power to generate the warning signals and the need for reliable and effective data. We demonstrate that techniques from data-intensive sciences can address these issues. Furthermore, we believe that the same approach, likely with additional computation, are applicable in the area of financial market cyber-security, which is widely acknowledged as important, but also largely ignored in the regulatory debate.

Acknowledgment

The authors would like to thank Horst Simon for his visionary leadership in establishing the Center for Innovative Financial Technology and initiating this project, David H. Bailey, Arie Shoshani, and Orianna DeMasi for helpful discussions in planning for this work, Harrison Dekker and Masou Nikravesh for helping with obtaining some of the data for this project, Jeff Donovan for the Nanex data used in this work, Marcos Lopez de Prado for providing sample code and data for computing VPIN, Maureen O’Hara and David Easley for valuable discussions on VPIN, Ananth Madhavan for instructive suggestions on computing HHI in a progressive manner, Quincey Koziol for helping out organizing HDF files, Dan Weisberg for useful discussion about Panopticon. We are grateful for the insightful discussions with Gregg Berman, SEC’s lead Flash Crash investigator, whose suggestions kick-started the current work.

This work is supported by the Director, Office of Laboratory Policy and Infrastructure Management of the U.S. Department of Energy under Contract No. AC02-05CH11231, and used resources of The National Energy Research Scientific Computing Center (NERSC).

References

[1] Bill Alpert and Lisa Stryjewski. Hitting the switch on new circuit breakers, Aug 2011. `http://su.pr/1kR67R`.

[2] E. Wes Bethel, Scott Campbell, Eli Dart, Kurt Stockinger, and Kesheng Wu. Accelerating Network Traffic Analysis Using Query-Driven Visualization. In *Proceedings of 2006 IEEE Symposium on Visual Analytics Science and Technology*, pages 115–122. IEEE Computer Society Press, October 2006. LBNL-59891.

[3] Graham Bowley. Ex-physicist leads flash crash inquiry. New York Times, Sep 2010. `http://www.nytimes.com/2010/09/21/business/economy/21flash.html`.

[4] CFTC/SEC. Findings regarding the market events of may 6,2010, Sep 2010. `http://www.sec.gov/news/studies/2010/marketevents-report.pdf`.

[5] CFTC/SEC. Preliminary findings regarding the market events of may 6,2010, May 2010. `www.sec.gov/sec-cftc-prelimreport.pdf`.

[6] Hank Childs, Eric S. Brugger, Kathleen S. Bonnell, Jeremy S Meredith, Mark Miller, Brad J Whitlock, and Nelson Max. A contract-based system for large data visualization. In *IEEE Visualization 2005*, pages 190–198, October 2005. Software available at `http://wci.llnl.gov/codes/visit/`.

[7] Jerry Chou, Kesheng Wu, Oliver Rübel, Mark Howison, Ji Qiang, Prabhat, Brian Austin, E. Wes Bethel, Rob D. Ryne, and Arie Shoshani. Parallel index and query for large scale data analysis. In *SC11*, 2011.

[8] B. Curry and K. D. George. Industrial concentration: A survey. *The Journal of Industrial Economics*, 31(3):203–255, 1983.

[9] D. Easley, N. Kiefer, M. O'Hara, and J. Paperman. Liquidity, information, and infrequently traded stocks. *Journal of Finance*, 51:1405–1436, 1996.

[10] David Easley, Marcos M. Lopez de Prado, and Maureen O'Hara. Flow Toxicity and Liquidity in a High Frequency World. Technical Report 9-2011, The Johnson School at Cornell University, Feb 2011.

[11] HDF Group. HDF5 user guide, 2011. `http://hdf.ncsa.uiuc.edu/HDF5/doc/H5.user.html`.

[12] Orris C. Herfindahl. *Concentration in the US Steel Industry*. PhD thesis, Columbia University, 1950.

[13] T. Hey, S. Tansley, and K. Tolle, editors. *The Fourth Paradigm: Data-Intensive Scientific Discovery*. Microsoft, Oct 2009.

[14] Albert O. Hirschman. *National Power and the Structure of Foreign Trade*. University of California Press, 1980.

[15] Mark Howison, Quincey Koziol, David Knaak, John Mainzer, and John Shalf. Tuning HDF5 for Lustre File Systems. In *Proceedings of 2010 Workshop on Interfaces and Abstractions for Scientific Data Storage (IASDS10)*, Heraklion, Crete, Greece, Sep 2010.

[16] Beni Lauterbach and Uri Ben-Zion. Stock market crashes and the performance of circuit breakers: Empirical evidence. *The Journal of Finance*, 48(5):1909–1925, Dec 1993.

[17] D. Leinweber. LBNL Center for Innovative Financial Technology. `http://www.lbl.gov/CS/CIFT.htm`.

[18] David Leinweber. Avoiding a billion dollar federal financial technology rat hole. *The Journal of Portfolio Management*, 37(3):1–2, 2011.

[19] David J. Leinweber. *Nerds on Wall Street: math, machines, and wired markets*. John Wiley and Sons, 2009.

[20] Anath Madhavan. Exchange-traded funds, market structure and the flash crash. Technical Report in-preparation, BlackRock, 2011.

[21] Alexis Madrigal. Market data firm spots the tracks of bizarre robot traders, Aug 2010. `http://su.pr/9l6seQ`.

[22] New York Stock Exchange, Inc. *TAQ 3 User's Guide (V 1.1.9)*, Oct 2008.

[23] M. O'Hara. *Market microstructure theory*. Blackwell, 2007.

[24] Oliver Rübel, Prabhat, Kesheng Wu, Hank Childs, Jeremy Meredith, Cameron G. R. Geddes, Estelle Cormier-Michel, Sean Ahern, Gunther H. weber, Peter Messmer, Hans Hagen, Bernd Hamann, and E. Wes Bethel. High Performance Multivariate Visual Data Exploration for Extemely Large Data. In *SuperComputing 2008 (SC08)*, Austin, Texas, USA, November 2008. LBNL-716E.

[25] SEC. SEC Proposes Consolidated Audit Trail System to Better Track Market Trades, SEC Press Release, May 26, 2010, `http://www.sec.gov/news/press/2010/2010-86.htm`.

[26] Kurt Stockinger, E. Wes Bethel, Scott Campbell, Eli Dart, , and Kesheng Wu. Detecting Distributed Scans Using High-Performance Query-Driven Visualization. In *SC '06: Proceedings of the 2006 ACM/IEEE Conference on High Performance Computing, Networking, Storage and Analysis*. IEEE Computer Society Press, November 2006. LBNL-60053.

[27] Kurt Stockinger, E. Wes Bethel, Scott Campbell, Eli Dart, and Kesheng Wu. Detecting distributed scans using high-performance query-driven visualization. In *SC '06*, page 82, New York, NY, USA, 2006. ACM.

[28] Avanidhar Subrahmanyam. Circuit breakers and market volatility: A theoretical perspective. *The Journal of Finance*, 49(1):237–254, Mar 1994.

[29] C. Wang, D. Chen, and M. Huang. International technology diffusion in computers & communications field. In *ICMIT*, pages 5–10, 2010.

[30] K Wu and et al. Fastbit: Interactively searching massive data. In *SciDAC 2009*, 2009. `http://www.iop.org/EJ/abstract/1742-6596/180/1/012053`.

[31] Kesheng Wu, Ekow Otoo, and Arie Shoshani. Optimizing bitmap indices with efficient compression. *ACM Transactions on Database Systems*, 31:1–38, 2006. `http://doi.acm.org/10.1145/1132863.1132864`.

[32] P. Yeh, W. Xia-Serafino, L. Miles, B. Kobler, and D. Menasce. Implementation of CCSDS lossless data compression in HDF. In *Earth Science Technology Conference-2002*, Jun 2002.