Enabling Data Analysis for Addressing Systemic Risk

Eric Hughes, PhD
Arnie Rosenthal, PhD
Charles Worrell, PhD

Abstract: Recently, the US experienced an economic crisis that shook confidence in key aspects of the financial system, and led to some calls for changes in the way the government tracks economic information that might warn of such a crisis. Among those changes was the creation of the Office of Financial Research (OFR), intended to collect and provide information to "anticipate emerging threats to financial stability or assess how shocks to one financial firm could impact the system as a whole" [OFR 2010]. These functions have been termed *systemic risk*: the risk that a threat to a large, single component of the financial system poses to the system as a whole, due to the inter-connectedness of the system and potential lack of consumer confidence in the system that might be caused if one component failed. This paper considers the computational approaches that may be needed in support of the mission of providing information about systemic risk, and possible mitigations of that risk. We acknowledge that there are many schools of thought for why the recent crisis occurred, the degree of systemic risk it posed, and possible government actions to mitigate the risk. Our position is that an agency such as the OFR with responsibility for monitoring systemic risk must be prepared to analyze diverse, uncertain information about the financial system and threats to it. Such an agency must be prepared to evaluate this information from multiple perspectives, and assess possible future outcomes given a variety of assumptions and regulatory responses.

Section 1: Computational Requirements

While researchers have discussed the data collection and handling needs of the Office of Financial Research (OFR) [OFR 2010], the computational and analytical needs have received less attention. The primary customer of OFR's information is the Financial Stability Oversight Council (FSOC). OFR is expected to develop standards for reporting of financial data, reducing uncertainty in this data and increasing transparency. OFR will use a Legal Entity Identifier (LEI) for each legally separable firm that engages in financial transactions. OFR was created with a large degree of independence [Schmidt 2010], to allow it to conduct analyses while minimizing influences of politics.

In some cases, a firm providing data to OFR is not incentivized to provide the best possible data for supporting OFR's mission. Providing this data may weaken the firm's position with respect to investors or competitors. The firm may need to collect additional data to provide to OFR, which may incur a substantial cost. The firm's personnel may not have the skills and background needed to provide good data to OFR. Data provided by the firm may embody different assumptions and interpretations of data standards than used by the OFR. As was demonstrated in the recent crisis, ratings from third-party organizations may not be reliable. As a result, OFR will need the capability to produce useful analyses from uncertain data, in some cases using multiple sources of data about the same firm or security.

OFR will collect and analyze information on the financial system, to assess the current and potential future states of the system, as well as explore potential government interventions and their expected effects on future states. This may require maintaining a (long) history of information about the financial system, as firms come and go and government regulation and policy evolves. The financial system is a complex system, and there are a wide variety of possible government responses to given situations. Any assessment of future states of the system will not be 100% certain, and the certainty of these assessments will depend on the quality of the data and analysis done. However, there is evidence from the recent crisis that at least one firm (Goldman-Sachs) had enough information to foresee the crisis, so it should be possible for OFR to anticipate some systemic risks.

This combination of requirements is not unique to OFR. Financial firms must analyze uncertain data about competitors and their own plans to evaluate strategies. Other government agencies are responsible for assessing other types of threats, using information from multiple sources. While the problem is not unique, expectations may be high, since OFR has been given broad power to collect data (albeit with significant privacy concerns), and the government is perceived as having significant ability to control the financial system.

Section 2: Overview of Analytic Approaches

There are several analytic approaches that might be used to assess systemic risk in the financial system based on data about the system. These approaches were designed to obtain actionable information from large, complex, noisy data, in a cost-effective manner. OFR might utilize all these approaches, or might focus on one or a few. Each approach requires different technology and skills, but they are all intended to be used by analysts, with expertise in information understanding, statistics, and deep understanding of the financial domain.

Section 2.1: Analytic Databases

Perhaps the most familiar and mature approach is to use a data warehouse implemented in a relational database management system (RDBMS). In these systems, incoming data is transformed into the data model of the warehouse via an Extract, Transform and Load (ETL) process. The ETL process can be used to cleanse data, addressing data quality issues that might prohibit successful analysis. ETL can also be used to tag data with source and other metadata. Often, detailed data records in the warehouse are aggregated or summarized into data marts (also RDBMSes) that are used by data mining or statistical analysis tools. The warehouse and marts can be queried using Structured Query Language (SQL). When text or other unstructured data are included, they are either processed and analyzed separately, or structured information is extracted from them and loaded into the warehouse. All phases of this process are currently supported by commercial technologies. This approach is shown in Figure 2.1.

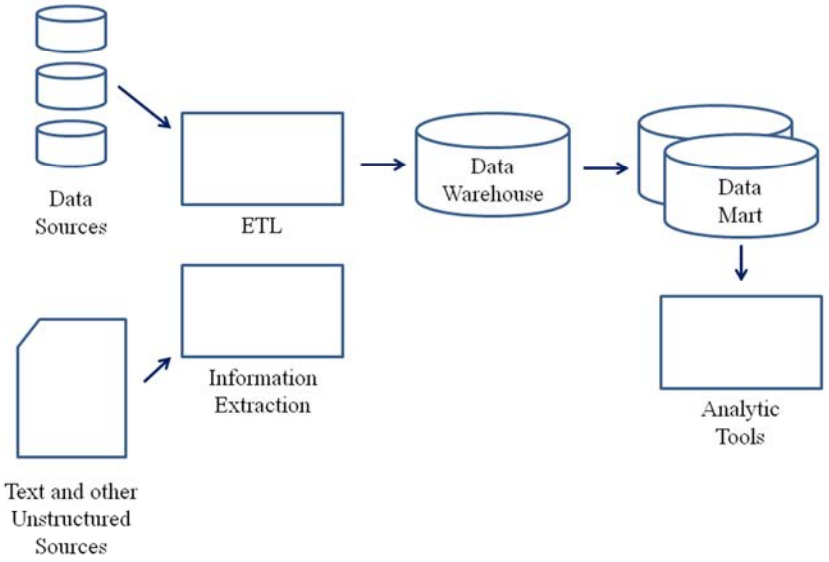

Figure 2.1: Relational database approach

Section 2.2: Parallel Databases

In the classic database approach, it can be difficult and expensive to scale with growing data sizes, and to cope with the need for ever more complex analytics run against data that is constantly arriving. One response to these needs is a growing trend toward massively parallel processing (MPP) databases. In an MPP database, data is sharded, or distributed across a cluster of processing nodes, each with storage and compute capability. There are several key differences between MPP databases and classic databases:

1. As data size grows, the MPP database can be scaled out with additional hardware more predictably.
2. Rather than using indexes and highly optimized queries to achieve performance, MPP databases use parallelism. This saves the need to store and update indexes and simplifies the process to add a new type of data to the system.
3. The MPP database requires a sharding function that spreads data and computation across the array to successfully use parallelism.
4. Often data can be loaded into the MPP database in parallel. In addition, data can be transformed and cleansed after it is loaded (ELT) or both before and after loaded (ETLT), taking advantage of the parallelism and consistency checking provided by the database while possibly creating additional copies.
5. Data marts may not be needed; analytics can often be done directly on the warehouse. Some analytic tools can push computation into the warehouse to save on I/O and data movement.

The MPP database approach is shown in Figure 2.2.

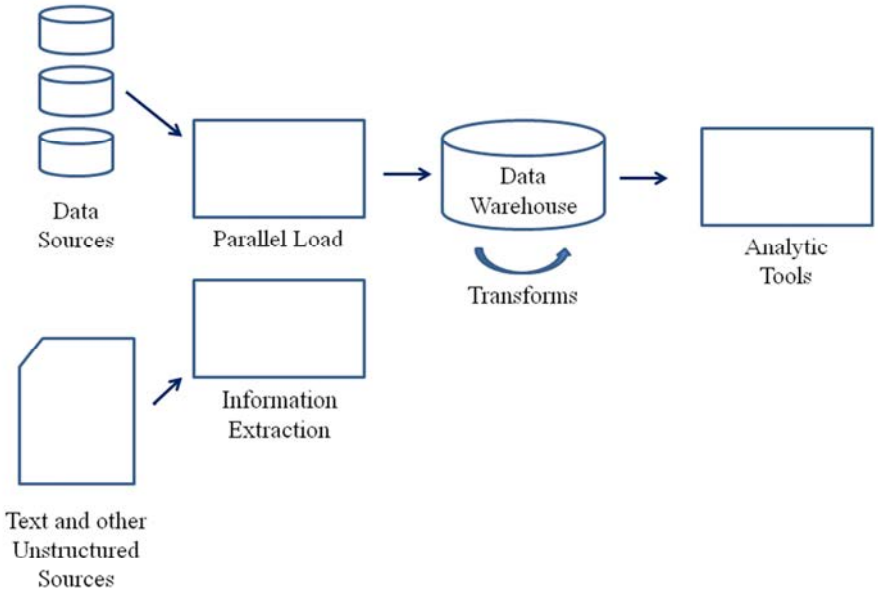

Figure 2.2: MPP database approach

Section 2.3: NoSQL Databases

Another growing trend in analytic databases is to relax some guarantees on atomicity or isolation of transactions, and drop native support for SQL queries. These databases are referred to as NoSQL databases, which we interpret as “Not Only SQL” databases, since add-on capabilities are often used to emulate subsets of SQL. NoSQL databases support a wide variety of data models, including semantic web triples, graphs / networks, and semi-structured documents. We focus on one particular variety of NoSQL database, where the data model is the key-value model, which differs from the relational model in several important ways:

1. Instead of a pre-declared schema, new attributes can be added at any time. These are often grouped in column families, allowing efficient storage and analytics.
2. All records have a timestamp and are versioned. Instead of updating a record in place, a new version is created. Analytics use only the most current version for a given key, by default, or can access prior versions.
3. Like MPP databases, the data is often sharded across an array of processing units. Indexes are typically not supported as opposed to less frequently used in MPP databases. The warehouse can be scaled out as needed. New data types can be added as needed.
4. Tables are stored in sorted order, by key. This supports fast retrieval of a record by key, or of a range of keys.
5. SQL is typically only supported by add-on tools.

The NoSQL database approach is shown in Figure 2.3.

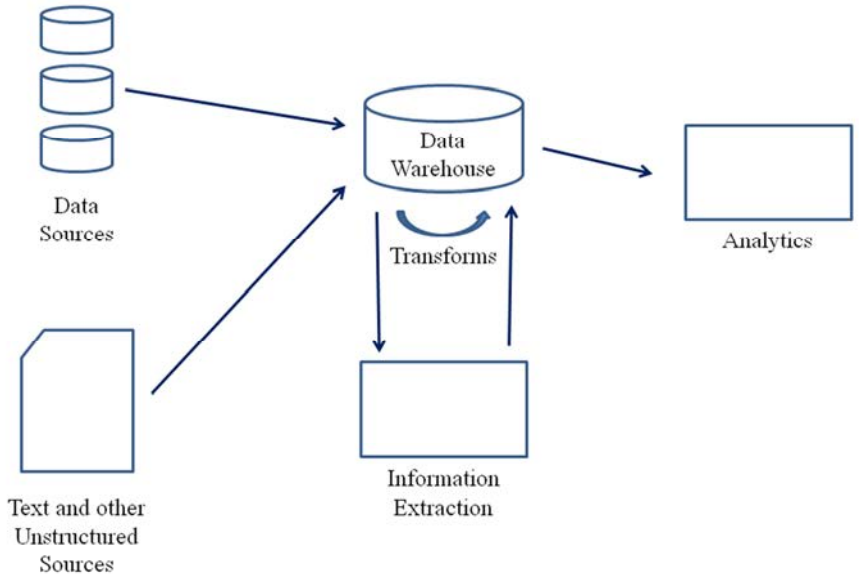

Figure 2.3: NoSQL database approach

One reason the NoSQL database approach was developed for complex data analytics is that it does not require ETL as a precursor step to making data available for analytics, because new data types can be added at any time so data need not be transformed to a common data model. This allows analytics to be done on a new type of data very quickly, with the risk that the new data may not be cleansed, integrated or even understood sufficiently to support sound analysis. Transformations can still be done in a NoSQL system, either before or after data is loaded into the database. Most users of NoSQL databases are currently using open source software, though this may reflect the early state of evolution of the NoSQL industry. Users of this technology currently build custom analytics using approaches like key-value lookup, MapReduce, and perhaps SQL. If desired, data from the warehouse can be loaded into data marts (relational databases that support SQL) that enable use of commercial analytic tools.

Many NoSQL databases lack mature security and access control capabilities. A notable exception is the Accumulo database, a government-developed, key-value NoSQL database that was released into the open source community in 2011 [NSA 2011]. Accumulo is modeled after Google's BigTable, and is comparable to HBase and other key-value NoSQL databases. Accumulo supports an information security approach where each data record (or potentially, cell) is labeled with visibility attributes that govern access to the data. The visibility attributes are then used in combination with user certificates and enterprise attributes (e.g., roles of each user) to manage access to the data, at the desired granularity.

Section 2.4: Semantic Web

Some have argued that systemic risk assessment would benefit from semantic web data models and reasoning [NSF]. Commercial and open source DBMSes are available that support semantic web models. We refer to these as semantic triple stores. Semantic triple stores represent data as triples, of the form:

Subject: Relationship => Object

For example, a financial firm (subject) might offer (relationship) a given financial instrument (object). To represent the price at which the instrument is offered, one might reify the offer relationship:

Financial Firm: Has-Relationship => Offer Relationship
Offer Relationship: Has-Object => Instrument
Offer Relationship: Has-Price => Price

Recently, semantic triples stores have been demonstrated with up to 1 billion triples (see for example published claims by Franz's AllegroGraph, and the Billion Triple Challenge). However, doing reasoning on triple stores of this size remains a research challenge. These stores are being explored to support graph analytics like social network analysis, since the intrinsic data model naturally supports vertices (subjects and objects) and edges (relationships).

Section 2.5: Analytic Cloud Computing

Some analytics efforts have abandoned databases all together, often in attempts to analyze massive amounts of data in a cost effective manner. In the most basic form of analytic cloud computing, data is stored in a distributed file system in an array, and a parallel program written is used to perform data processing. Hadoop is an open source implementation of this approach. Hadoop includes two key components:

- The Hadoop Distributed File System (HDFS), which gives the view of a single file system, implemented over an array of servers (each with its own storage). HDFS automatically replicates file blocks for fault tolerance. HDFS is designed to manage very large files, using large block size for greater disk I/O throughput.
- MapReduce is a simple language for writing programs that execute in parallel across the array. In the first step, a Map function is applied to each object in a set, producing an intermediate result set. The intermediate result set is shuffled and written to disk in the array. Then, a Reduce function is used to aggregate final results from the intermediate results.

Hadoop manages the sharding of data in the array. Each Map and Reduce step is executed by a number of Map and Reduce processes, each working with a subset of the data. These processes can work in parallel without inter-process communication, which greatly simplifies the task of writing a correct parallel program. Hadoop manages the startup, execution and completion of these processes, and deals with faults that occur by starting new Map or Reduce processes, perhaps using replicated copies of the data. In essence, Hadoop supports parallel analytics that work over massive data, in a way that is tolerant of faults that occur during long-running programs. A typical analytic consists of a sequence of MapReduce jobs, each taking as input the output of the prior job.

Hadoop and MapReduce are often used in combination with key-value NoSQL databases. In fact, many key-value NoSQL databases use HDFS for file management. The resulting

analytic system supports MapReduce parallel programs running on underlying files or key-value tables. These systems also use key-value tables to pre-compute complex analytics (akin to materialized views in traditional databases). MapReduce can be used to cleanse or transform data as it is ingested into the system, and as analytics are pre-computed in key-value tables. MapReduce can also be used to compute transformed data sets that can be fed to analytic tools or other databases. A variant of the analytic cloud approach is shown in Figure 2.4.

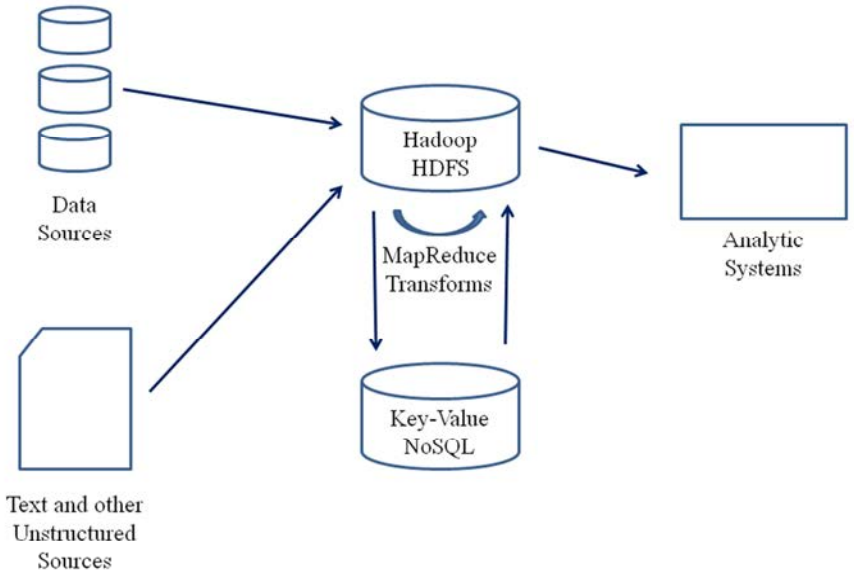

Figure 2.4: Analytic cloud approach

Section 2.6: Complex Event Processing

The approaches we have discussed so far are designed to support complex analytics on massive amounts of data. If data is continuously arriving, they typically ingest data in batches, where data is cached until enough has been received or a time period has elapsed, then the data is ingested into the system. This approach can be described as high throughput, but potentially high latency, meaning that massive amounts of data can be processed by the system over time, but the time from arrival of a new data object until analytic results are first available that use the new data can be long.

Although efforts are underway to reduce this latency for database approaches, an alternative is to not store the data persistently. Complex event processing (also referred to as stream mining) has been developed specifically for problems where massive data is continuously arriving and analytics are primarily used on most recent data and aggregates over time windows. In this approach, data is cached temporarily but not persistently stored. Analytics or queries are stored persistently and executed continuously, rather than run once as is typical for database approaches. Notionally, as data streams through the

system, analytics are applied to identify patterns of interest and route data to other systems. These other systems can include analytic systems, and any of the previously discussed database and analytic cloud approaches. This approach is shown in Figure 2.5. In this figure, the complex event processing system performs “Stream Analytics” on incoming streams of text or structured data. Alerts are generated for threshold crossing and other low-latency analytic results. Some data is selected to feed into a more traditional data warehouse; the rest is discarded.

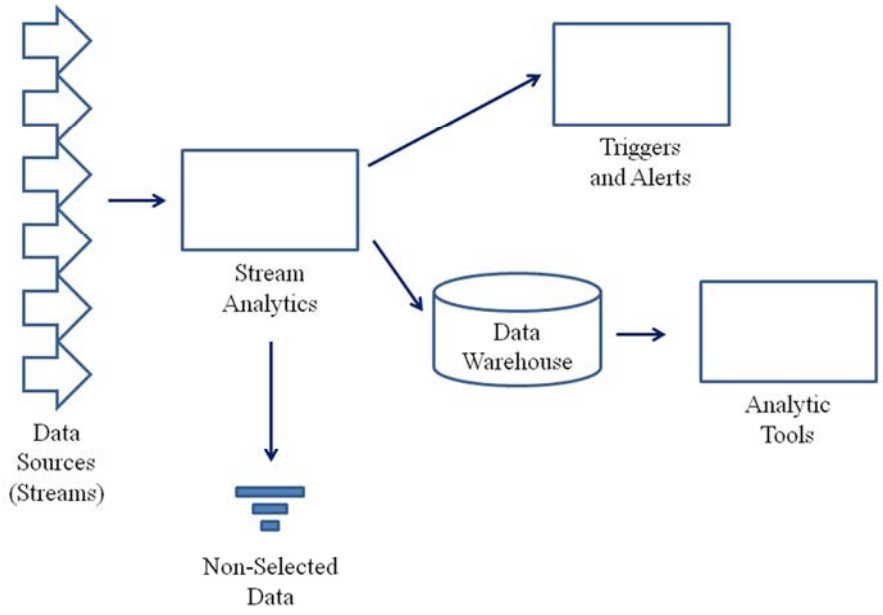

Figure 2.5: Complex event processing approach

Section 3: Analysis Approaches

We have discussed some approaches for preparing data for analysis, and some of the basic means of analysis they support (e.g., SQL queries). In this section, we explore perhaps the most important analysis approach for the systemic risk community: modeling and simulation. We also give a brief overview of multi-party computation, which has been suggested for use in assessing systemic risk.

Section 3.1: Modeling and Simulation

Here we discuss approaches that rely on models of the financial system, rather than focusing on data about the system. There are models that are recognized to represent the way that portions of the financial system work (e.g., European Central Bank (2009, 2010)) and these models sometimes allow us to perform high confidence analyses with great efficiency. There are also times when a lack of data forces us to rely on approaches that may be rule driven (heuristic), probability driven (stochastic), or behavior driven

(agent-based). Lack of data may result from a change in policy or business climate that results in previous records no longer being applicable to the current environment (such as the repeal of the Glass-Steagall Act in 1999), or from the recognition that an important factor in the analysis has not previously been measured.

In addition, when conjecture about future conditions is required, our models of how elements of the financial system interact may allow us to forecast with more confidence than we could by only relying on trends embedded in data from the past. Lastly, what-if analyses that explore the potential impacts of various policy interventions under consideration may require the use of models that can support an analysis by generating their own data about events that never actually happened, through simulation.

Heuristic models such as Principal Components Analysis (PCA) use Granger causality to derive linkages and interconnections within the financial system that lead to contagion risk from measures of correlation between equity returns Billio, et al. (2010). The PCA method represents the financial system as four interacting sectors: banks, brokers, insurance companies, and hedge funds.

Stochastic models like Systemic Contingent Claims Analysis (CCA) have been used to estimate the government's contingent liabilities in the event of a systemic crisis [Gray and Jobst (2010)]. The CCA method was originally developed for firm-level risk management using stochastic processes and has since been extended to evaluating systemic risk.

Agent based models similar to the Zero Based Intelligence model (Farmer, et al. (2005)) use a simulation of the behavior of a community of agents, following their separate motivations, to demonstrate market order-book structure and price behavior with respect to randomly placed (zero intelligence) trades.

Most of these model types can be used in the analysis of the same data organized under a range of computational approaches, as described in this paper.

Section 3.2: Managing Model Runs

There are many competing models used in financial analysis. In addition, new models will always be needed, as the financial system evolves, and as we learn more about it and the associated systemic risks. Thus, use of models to assess systemic risk requires an ability to manage the runs of models, so the *provenance* of model results can be tracked and their results can be used to make assessments.

A given model may take variety of input parameters, including time period simulated and other aspects under the analyst's control. These inputs must be tracked as part of the provenance of model results. Automatic capture of this information is strongly preferred, for traceability. While simulation tools may include this capability, each is likely to track provenance in its own way – standards and translators will be needed.

Section 3.3: Multi-Party Computation

The approaches we have discussed so far would typically be used within an organization. Addressing systemic risk will require cooperation between multiple organizations, including government agencies and financial firms. Although laws will require sharing of some sensitive information to address systemic risk, each organization may have relevant data that cannot be shared, and may have unique capabilities to analyze the data available within the organization. For this reason, researchers have been developing a theory of multi-party computation, in which parties do not need to share their most sensitive data.

In multi-party computation, the first step is to identify what data can and cannot be shared. Data that can be shared can also be used to coordinate the multi-party computation. In some cases, raw data cannot be shared, but aggregates or masked data can. Algorithms have been developed in which the computation can proceed with minimal data sharing. Sometimes the computation is asymmetric; where different parties (e.g., bank and government) contribute different sorts of data. Other times it is symmetric; for example, a group of financial firms may want to assess whether they jointly perceive a systemic risk in the financial services they offer, without disclosing the data they individually use to assess risk. A multi-party computational approach partitions the assessment into analytics that each organization can execute privately.

Many multi-party algorithms have been developed, and shown to have good security properties. However, many of these approaches are fragile, no longer working if one alters the problem formulation slightly. Organizations in a multi-party computation could each use their own preferred computational approach, or could use a shared approach. In some cases, a neutral third party is used to perform the multi-party computation and distribute appropriate results to each participating organization.

Section 3.4: Information Security

The data used to assess systemic risk, and the resulting assessments, will be sensitive and must be protected from disclosure, corruption and theft. While it is beyond our scope to give a full description of all the steps required to protect information, here we touch on two aspects: labeling of data, and sensitivity of analysis results.

The approaches we have discussed vary in their ability to label data records with sensitivity metadata. There are many possibilities for this metadata: disclosure may be limited by law or policy, retention may be limited by law or policy, access may require verification of a certain level of trust in the person or system accessing, etc. In addition, a given piece of information may be highly sensitive for one period of time, then less sensitive afterwards. In some cases, sensitivity metadata might need to be tracked at a fine granularity – for individual records or even values (cells). Most commercial database systems support sensitivity metadata at the table level, and can emulate it at the record or value level (with significant performance impact and space overhead). Accumulo also supports sensitivity metadata down to the value level.

It might seem that the sensitivity of an analysis result could be determined as the maximum of the sensitivities of all the inputs that went into the result. However, this approach has two well-known limitations:

1) Sensitivity of the result might be over-estimated if a highly sensitive input doesn't really affect the result
2) Sensitivity of the result might be under-estimated if the analysis creates new information beyond the straight-forward combination of the inputs – often, the whole is greater than the sum of the parts

Despite these limitations, it is important that analysis results have sensitivity metadata. If a human analyst is involved in creating the results, the human should have some input to the sensitivity of the result. Where necessary, results might be tentatively labeled based on the maximum of the sensitivities of the inputs. In any case, it should be recognized that sensitivity of analysis results may be less certain than that of data inputs.

Information security will likely be an essential element of a successful approach for assessing systemic risk. Firms asked to provide sensitive data will need a high degree of trust in the information security of the approaches used.

Section 4: Discussion and Future Work

We have presented several approaches for assessing systemic financial risk, using large amounts of data in a variety of formats that are constantly arriving. Some approaches can be used in combination. Choosing the right approach or combination of approaches is a complex systems engineering task, involving a deep understanding of the types of data available, the kinds of analyses to be conducted, and the policies for protecting and sharing sensitive data. We believe the systems engineer needs to be versed in a variety of approaches – choosing the right combination depends on a clear understanding of what data will be used, and what kinds of analysis will be performed.

One challenge is to deal with ever-increasing volumes of data. Many of the approaches we have discussed aspire to scale linearly in the amount of data. In the ideal, this means that a given computation can be performed on twice as much data in the same amount of time, by using twice as much hardware. For some situations, the analytic cloud, MPP database and NoSQL database approaches in particular have been shown to have near-linear scalability. Sometimes this near-linear scalability is limited by the network bandwidth available within the array of computing resources used. One important opportunity for future research is to identify (in ways that they can be anticipated) and overcome the theoretical and practical break points in near-linear scalability for different approaches.

We expect that assessment of systemic risk will require the use of new types of data and new analytics over time, as the global financial services industry evolves and as experience is gained in assessing systemic risk. Some approaches (analytic cloud, MPP database, NoSQL database) achieve a high level of agility by deferring the data cleansing and performance tuning used in a traditional relational database analytic approach. Instead, they use brute-force parallelism and simplified programming models to allow an

analytic effort to keep pace with rapidly-evolving problems. Data cleansing needs to go beyond simple value checks, e.g., to identify systematic biases. Data quality is not primarily technical, however – it is primarily about monitoring quality of what is supplied, setting priorities on what needs to be improved, and giving the provider incentives and feedback to improve what they supply. Another research opportunity is to extend these approaches to continuously cleanse and improve the data they use, and dynamically tune for performance as analysis access patterns emerge. In a sense, this would give the best of both worlds – new types of data could be exploited immediately, with strong caveats about the quality of the results, and over time further exploited with fewer caveats.

The computational approaches we have described can support complex analytics over massive, varied data. The results of these analytics can be very difficult for the human analyst to understand, involving deep knowledge of the semantics of the data, and detailed information about how the analytic was implemented. Some researchers are investigating techniques for explaining the results of data mining algorithms to users, but we believe additional research is needed, especially for analytic approaches that do not use an integrated data model and do less cleansing of data.

We discussed multi-party computation as a way to manage complex analytics over sensitive data. Multi-party computation is an active area of research. We also see many other opportunities for research in information security for analytic computational approaches. For example, these approaches are often designed to work across a diverse assortment of data sets, each of which may have different sensitivity or restrictions on sharing. One approach is to prevent computations from revealing sensitive information (even by inference); however, this is extraordinarily difficult. Alternatively, more feasible but less directly meeting security goals, one can limit what data is used by or visible to whom.

We need techniques for managing which analytics can be performed on which data, to demonstrably ensure that the analytics adhere to data protection or usage policies. We also need techniques for deriving security attributes for analytic results, so that systems can automatically determine which users can see these results, and system behavior can be audited and shown to adhere to policy. While these research areas have long been explored for traditional relational databases, there is need to revise and apply them to the full range of approaches we have discussed. We also see a need for research in approaches to mask or anonymize sensitive data for some uses, including development of new analytics and research into new analytic techniques.

In our view, one of the more difficult aspects of assessing systemic financial risk is dealing with the role of consumer confidence, or more generally, people's perceptions of systemic risk and how government agencies are responding to it. For example, much has been written about financial firms that are perceived to be "too big to fail", and the related expectation that government will bail out such firms if they falter. If understanding these issues is essential to assessing systemic risk, then the models need to include a variety of data, including social media and other sources for consumer

confidence (which are structurally and semantically very different from traditional financial data sources). While researchers are currently addressing socio-cultural modeling and "smart power", we see many opportunities to apply this to systemic risk and other financial analytic problems. Supporting such models will lead to a new generation of computational challenges.

In a free market society, the challenge often involves finding a balance between regulation and freedom to innovate and compete. Much has been written about how some innovations in mortgage securities created systemic risks in the recent financial crisis. We see a need for analytic models that account for innovation in financial instruments and that include potential government responses other than just restrictions on innovation. These models should also have relevance in a world where there are multiple governance philosophies, including both national and international (e.g., the European Union). We see need for models that can help government agencies cope with financial innovation, while addressing systemic risk in a globally-connected economy. Addressing this grand challenge will require multi-disciplinary research, using a variety of computational approaches on large, complex data.

References:

NSA (2011). Accumulo open source software. See https://wiki.apache.org/incubator/AccumuloProposal.

Billio, M., Getmansky, M., Lo, M. and Pelizzon, L. (2010). Econometric measures of systemic risk in the finance and insurance sectors. *National Bureau of Economic Research Working Paper 16223*.

European Central Bank (2010). New quantitative measures of systemic risk. *Financial Stability Review Special Feature E*.

European Central Bank (2009), Recent advances in modeling (sic) systemic risk using network analysis", *Workshop summary*.

Farmer, J., Patelli, P. and Zovko, I. (2005). The predictive power of zero intelligence in financial markets. *Proceedings of the National Academy of Science*, **102**, pp. 2254-2259.

Gray, D. and Jobst, A. (2010). Systemic CCA - a model approach to systemic risk. *Conference sponsored by the Deutsche Bundesbank and Technische Universitaet Dresden*.

OFR (2010). Office of Financial Research created under the Dodd-Frank Wall Street reform and consumer protection act: Frequently Asked Questions. See http://www.treasury.gov/initiatives/Documents/OFR_FAQ-11242010-FINAL.PDF .

Schmidt, R. (2010). The Treasury's new research office. *Business Week*.