Fraud Detection: Discovering Connections with Graph Databases

WHITE PAPER

Gorka Sadowksi & Philip Rathle

White Paper

TABLE OF CONTENTS

Fraud Detection: Discovering Connections Using Graph Databases

Gorka Sadowksi & Philip Rathle

Introduction

Banks and Insurance companies lose billions of dollars every year to fraud. Traditional methods of fraud detection play an important role in minimizing these losses. However increasingly sophisticated fraudsters have developed a variety of ways to elude discovery, both by working together, and by leveraging various other means of constructing false identities.

Graph databases offer new methods of uncovering fraud rings and other sophisticated scams with a high-level of accuracy, and are capable of stopping advanced fraud scenarios in real-time.

While no fraud prevention measures can ever be perfect, significant opportunity for improvement can be achieved by looking beyond the individual data points, to the connections that link them. Oftentimes these connections go unnoticed until it is too late — something that is unfortunate, as these connections oftentimes hold the best clues.

Understanding the connections between data, and deriving meaning from these links, doesn't necessarily mean gathering new data. Significant insights can be drawn from one's existing data, simply by reframing the problem and looking at it in a new way: as a graph.

Unlike most other ways of looking at data, graphs are designed to express relatedness. Graph databases can uncover patterns that are difficult to detect using traditional representations such as tables. An increasing number of companies are using graph databases to solve a variety of connected data problems, including fraud detection.

This paper discusses some of the common patterns that appear in three of the most damaging types of fraud: first-party bank fraud, insurance fraud, and e-commerce fraud. While these are three entirely different types of fraud, they all hold one very important thing in common: the deception relies upon layers of indirection that can be uncovered through connected analysis. In each of these examples, graph databases offer a significant opportunity to augment one's existing methods of fraud detection, making evasion substantially more difficult.

Example 1: First-Party Bank Fraud

Background

First-party fraud involves fraudsters who apply for credit cards, loans, overdrafts and unsecured banking credit lines, with no intention of paying them back. It is a serious problem for banking institutions.

U.S. banks lose tens of billions of dollars every year[1] to first-party fraud, which is estimated account for as much as one-quarter or more of total consumer credit chargeoffs in the United States.[2] It is further estimated that 10%-20% of unsecured bad debt at leading US and European banks is misclassified, and is actually first party fraud.[3]

The surprising magnitude of these losses is likely the result of two factors. The first is that first-party fraud is very difficult to detect. Fraudsters behave very similarly to legitimate customers, until the moment they "bust out", cleaning out all their accounts and promptly disappearing.

A second factor— which will also be explored later in greater detail—is the exponential nature of the relationship between the number of participants in the fraud ring and the overall dollar value controlled by the operation. This connected explosion is a feature often exploited by organized crime.

However while this characteristic makes these schemes potentially very damaging, it also renders them particularly susceptible to graph-based methods of fraud detection.

Typical Scenario

While the exact details behind each first-party fraud collusion vary from operation to operation, the pattern below illustrates how fraud rings commonly operate:

1. A group of two or more people organize into a fraud ring
2. The ring shares a subset of legitimate contact information, for example phone numbers and addresses, combining them to create a number of synthetic identities
3. Ring members open accounts using these synthetic identities
4. New accounts are added to the original ones: unsecured credit lines, credit cards, overdraft protection, personal loans, etc.
5. The accounts are used normally, with regular purchases and timely payments
6. Banks increase the revolving credit lines over time, due to the observed responsible credit behavior
7. One day the ring "busts out", coordinating their activity, maxing out all of their credit lines, and disappearing
8. Sometimes fraudsters will go a step further and bring all of their balances to zero using fake checks immediately before the prior step, doubling the damage
9. Collections processes ensue, but agents are never able to reach the fraudster
10. The uncollectible debt is written off

1. Experian at http://www.experian.com/assets/decision-analytics/whitepapers/first-partyfraud-wp.pdf
2. Experian at http://www.experian.com/assets/decision-analytics/whitepapers/first-partyfraud-wp.pdf
3. Business Insider 2011 at http://www.businessinsider.com/how-to-usesocial-networks-in-the-fight-against-first-party-fraud-2011-3

In order to illustrate this scenario, let's take a (small) ring of 2 people colluding to create synthetic identities:

1. Tony Bee lives at 123 NW 1st Street, San Francisco, CA 94101 (his real address) and gets a prepaid phone at 415-123-4567
2. Paul Favre lives at 987 SW 1st Ave, San Francisco, CA 94102 (his real address) and gets a prepaid phone at 415-987-6543

Sharing only phone number and address (2 pieces of data), they can combine these to create 22 = 4 synthetic identities with fake names as described in Diagram 1 below.

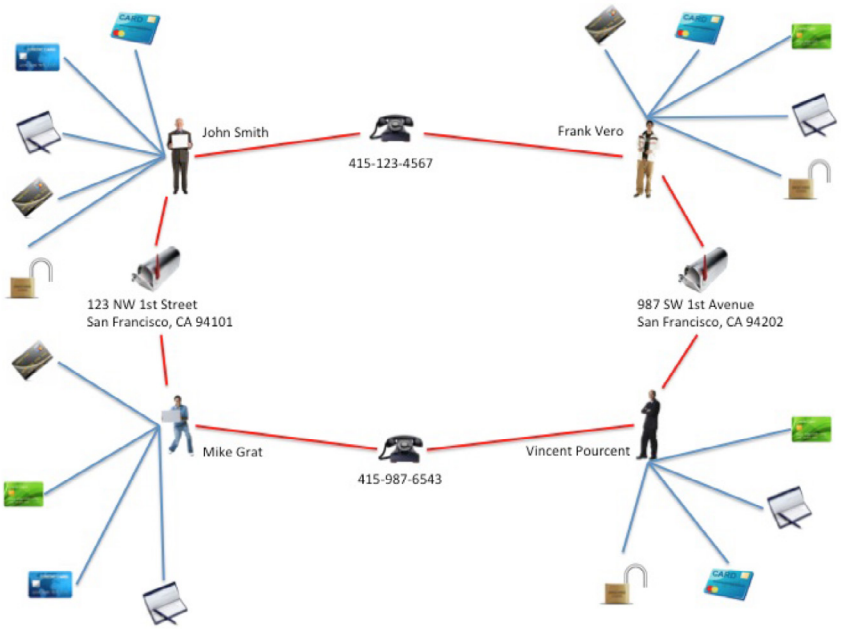

Diagram 1: 2 people sharing 2 pieces of data and creating 4 synthetic identities

Diagram 1 shows the resulting fraud ring, with 4-5 accounts for each synthetic identity, totaling 18 total accounts. Assuming an average of $4K in credit exposure per account, the bank's loss could be as high as $72K.

As in the process outlined above, the phone numbers are dropped after the bust-out, and when the investigators come to these addresses, both Tony Bee and Paul Fabre (the fraudsters, who really live there) deny ever knowing John Smith, Frank Vero, Mike Grat or Vincent Pourcent.

Detecting the Crime

Catching fraud rings and stopping them before they cause damage is a challenge. One reason for the challenge is that traditional methods of fraud detection are either not geared to look for the right thing: in this case, the rings created by shared identifiers. Standard instruments—such as a deviation from normal purchasing patterns— use discrete data and not connections. Discrete methods are useful for catching fraudsters acting alone, but they fall short in their ability to detect rings. Further, many such methods are prone to false positives, which creates undesired side effects in customer satisfaction and lost revenue opportunity.

Gartner proposes a layered model for fraud prevention,[4] which can be seen below:

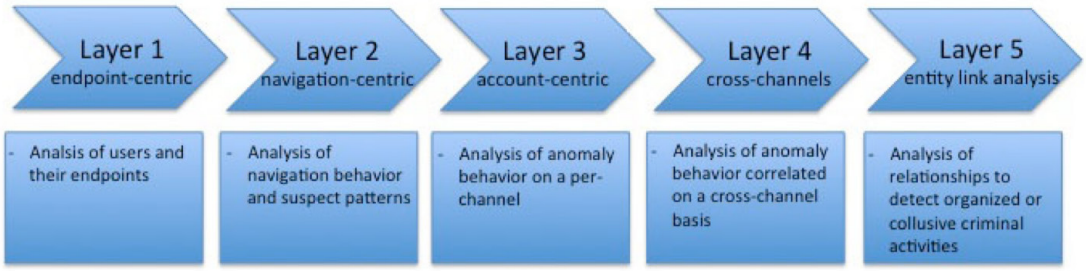

Diagram 2: Gartner's Layered Fraud Prevention Approach

It starts with simple discrete methods (at the left), and progresses to more elaborate "big picture" types of analysis. The rightmost layer, "Entity Link Analysis", leverages connected data in order to detect organized fraud. As will be shown in the following sections, collusions of the type described above can be very easily uncovered—with a very high probability of accuracy—using a graph database to carry out entity link analysis at key points in the customer lifecycle.

Entity Link Analysis

We discussed earlier how fraudsters use multiple identities to increase the overall size of their criminal takings. It's not just the dollar value of the impact that increases as the fraud ring grows, it's also the computational complexity required to catch the ring. The full magnitude of this problem becomes clear as one considers the combinatorial explosion that occurs as the ring grows. In the diagram below, one can see how adding a third person to the ring expands the number of synthetic identities to nine:

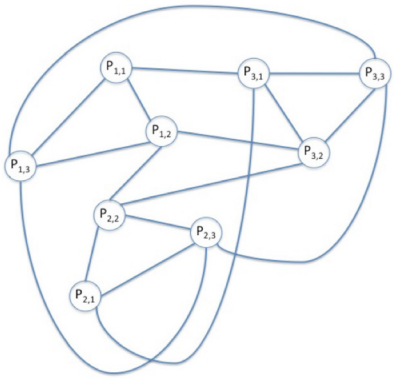

Diagram 3: 3 people each sharing 2 valid identifiers results in 9 interconnected synthetic identities
A ring of n people (n≥2) sharing m elements of data (such as name, date of birth, phone number, address, SSN, etc.) can create up to n^m synthetic identities, where each synthetic identity (represented as a node) is linked to m × (n-1) other nodes, for a total of (n^m × m × (n-1)) / 2 relationships.

Likewise, four people can control 16 identities, and so on. The potential loss in a ten-person fraud bust-out is $1.5M, assuming 100 false identities and 3 financial instruments per identity, each with a $5K credit limit.

4. Gartner at http://www.gartner.com/newsroom/id/1695014

How Graph Databases Can Help

Uncovering rings with traditional relational database technologies requires modeling the graph above as a set of tables and columns, and then carrying out a series of complex joins and self joins. Such queries are very complex to build and expensive to run. Scaling them in a way that supports real-time access poses significant technical challenges, with performance becoming exponentially worse not only as the size of the ring increases, but as the total data set grows.[5]

Graph databases have emerged as an ideal tool for overcoming these hurdles. Languages like Cypher provide a simple semantic for detecting rings in the graph, navigating connections in memory, in real time.[6]

The graph data model in Diagram 4 below represents how the data actually looks to the graph database, and illustrates how one can find rings by simply walking the graph:

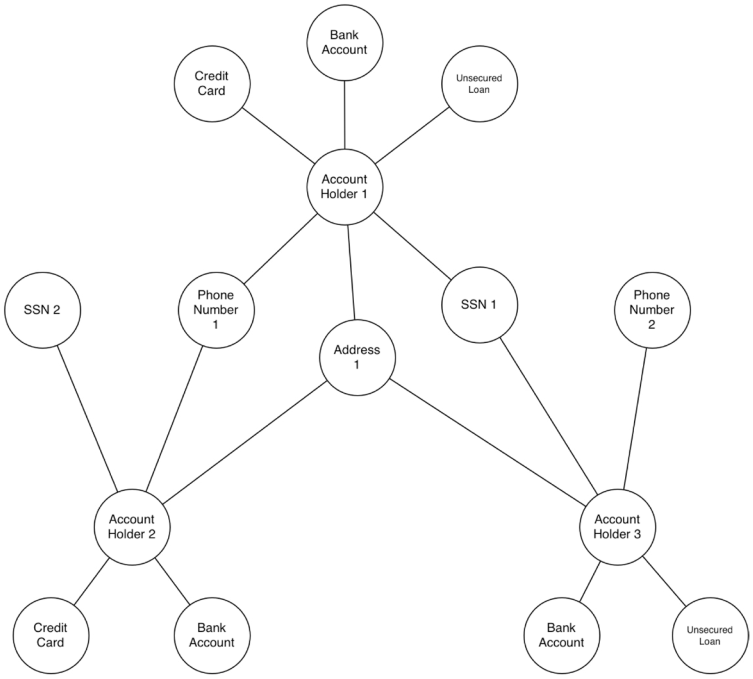

Diagram 4: Subset of a Fraud Ring, As Modeled in a Graph Database

Augmenting one's existing fraud detection infrastructure to support ring detection can be done by running appropriate entity link analysis queries using a graph database, and running checks during key stages in the customer & account lifecycle, such as:

1. at the time the account is created,
2. during an investigation,
3. as soon as a credit balance threshold is hit, or
4. when a check is bounced.

Real-time graph traversals tied to the right kinds of events can help banks identify probable fraud rings: during or even before the Bust-Out occurs.

5. Graph Databases, O'Reilly, Ian Robinson, Jim Webber & Emil Eifrem, Chapter 2 (ISBN: 978-1-449-35626-2)
6. Ibid. Pages 5 and 144

Example 2: Insurance Fraud

Background

The impact of fraud on the insurance industry is estimated to be $80 billion annually in the US, a number that has been growing in recent years.[7] From 2010 to 2012, questionable claims in the U.S. jumped 27 percent, to 116,171 claims in 2012, nearly half resulting from faked or exaggerated injury claims.[8] In the UK, insurers estimate that bogus whiplash claims add $144 per year to each driver's policy.[9]

Insurance fraud attracts sophisticated criminal rings who are often very effective in circumventing fraud detection measures, Once again, graph databases can be a powerful tool in combating collusive fraud.

Typical Scenario

In a typical hard fraud scenario, rings of fraudsters work together to stage fake accidents and claim soft tissue injuries. These fake accidents never really happen. They are "paper collisions", complete with fake drivers, fake passengers, fake pedestrians and even fake witnesses.

Because soft tissue injuries are easy to falsify, difficult to validate, and expensive to treat, they are a favorite among fraudsters, who have even developed a term for them "whiplash for cash".

Such rings normally include a number of roles.

1. Providers. Collusions typically involve participation from professionals in several categories:
    a. Doctors, who diagnose false injuries
    b. Lawyers, who file fraudulent claims, and
    c. Body shops, which misrepresent damage to cars
2. Participants. These are the people involved in the (false) accident, and normally include:
    a. Drivers
    b. Passengers
    c. Pedestrians
    d. Witnesses

Fraudsters often create and manage rings by "recycling" participants so as to stage many accidents. Thus one accident may have a particular person play the role of the driver. In another accident the same person may be a passenger or a pedestrian, and in another a witness. Clever usage of roles can generate a large number of costly fake accidents, even with a small number of participants.

In the scenario depicted in Diagram 5 on the following page, a six-person collusion results in three false accidents. Each person plays the role of "driver" once and "passenger" twice. Assuming an average claim of $20K per injured person, and $5K per car, the ring can claim $390K in total.

7. Coalition against insurance fraud at http://www.insurancefraud.org/article.htm?RecID=3274#.UnWuZ5E7ROA
8. National Insurance Crime Bureau at https://www.nicb.org/newsroom/news-releases/u-s--questionable-claims-report
9. Insurance Fraud Organization at http://www.insurancefraud.org/IFNS-detail.htm?key=17499#.UmmsJyQhZ0o

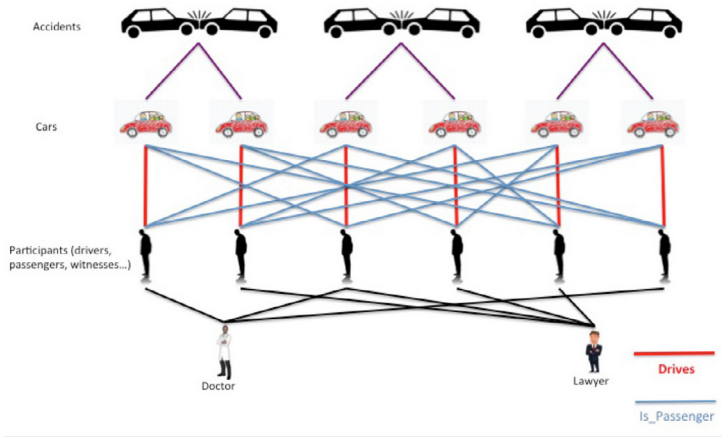

Diagram 5: Simple 6-people collusion

As in the earlier bank fraud example, the complexity and scale of such schemes can quickly soar. In an example where ten people collude to commit insurance fraud, five false accidents are staged, where each person plays the role of the driver once, a witness once and a passenger three times. Assuming an average claim of $40K per injured person and $5K per car, the ring can claim up to $1.6M for 40 people injured!

This example is depicted in Diagram 6 below:

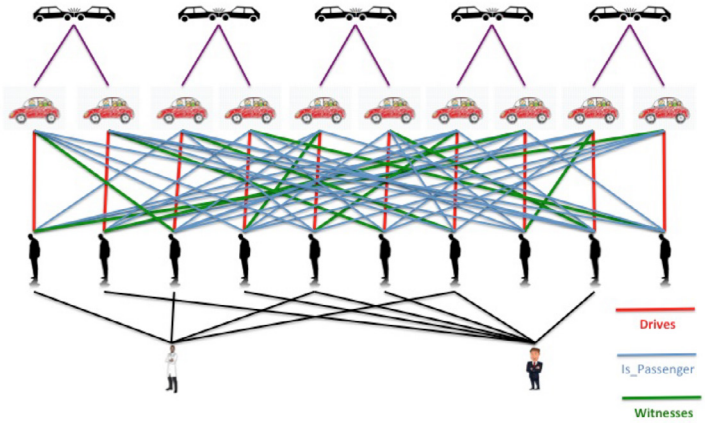

Diagram 6: Simple Ten-Person Collusion Depicted as a Graph

Detecting the Crime

As with bank fraud detection, a layered approach has emerged as a best practice for detecting insurance fraud. While existing analysis techniques are sufficient for catching certain fraud scenarios, sophisticated criminals often elude these methods by collaborating. Criminal rings are very skilled at concealing collusion, and inventing and staging complex "paper collisions" that do not arouse suspicion.

The next frontier in Insurance Fraud detection is to use social network analysis to uncover these rings. Connected analysis is capable of revealing relationships between people who are otherwise acting like perfect strangers.

How Graph Databases Can Help

As in the bank fraud example, social network analytics tends not to be a strength of relational databases. Discovering the ring requires joining a number of tables in a complex schema such as Accidents, Vehicles, Owners, Drivers, Passengers, Pedestrians, Witnesses, Providers, and joining these together multiple times— once per potential role— in order to uncover the full picture. Because such operations are so complex and costly, particularly for very large data sets, this crucial form of analysis is often overlooked.

On the other hand, finding fraud rings with a graph database becomes a simple question of walking the graph. Because graph databases are designed to query intricate connected networks, they can be used to identify fraud rings in a fairly straightforward fashion.

The graph below depicts how the above scenario might be modeled in a graph database:

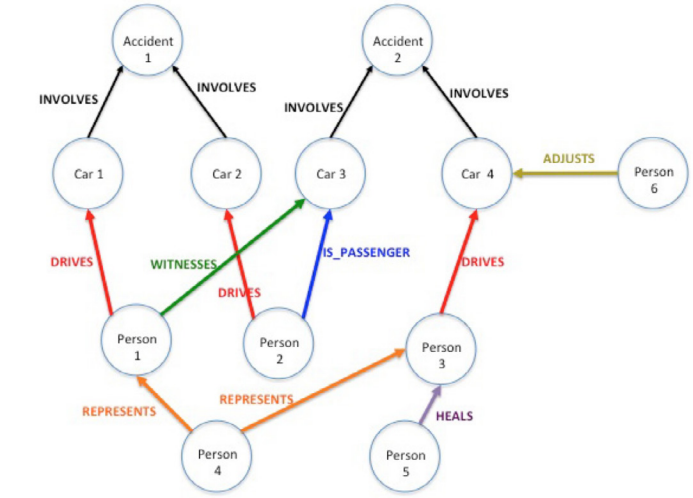

Diagram 7: A Graph Representation of Insurance Fraud

As in the Bank Fraud example above, graph database queries can be added to the insurance company's standard checks, at appropriate points in time—such as when the claim is filed—to flag suspected fraud rings in real time.

Example 3: e-Commerce Fraud

As our lives become increasingly digital, a growing number of financial transactions are conducted online. Fraudsters have been quick to adapt to this trend, and to devise clever ways to defraud online payment systems. While this type of activity can and does involve criminal rings, a well-educated fraudster can create a very large number of synthetic identities on his own, and use these to carry on sizeable schemes.

Fraud Detection: Discovering Connections with Graph Databases

Typical Scenario

Consider an online transaction with the following identifiers: user ID, IP address, geo location, a tracking cookie, and a credit card number. One would typically expect the relationships between these identifiers to be fairly close to one-to-one. Some variations are naturally to be tolerated to account for shared machines, families sharing a single credit card number, individuals using multiple computers, and the like. However as soon as the relationships begin to exceed a reasonable number, fraud is often at play.

For example: a large numbers of users may have transactions originating from the same IP, large numbers of shipments to different addresses may use the same credit card, or a large numbers of credit cards may all use the same address. In each of these scenarios, it is the pattern inside the graph—discovered by walking the relationships between disparate pieces of information—that can serve as strong indicating signals of a fraud event. The more interconnections exist amongst identifiers, the greater the cause for concern. Large and tightly-knit graphs are very strong indicators that fraud is taking place.

How Graph Databases Can Help

As in the first-party bank fraud, and insurance fraud examples above, graph databases are designed to carry out pattern discovery in real time across precisely these kinds of data sets. By putting checks into place and associating them with the appropriate event triggers, such schemes can be uncovered before they are able to inflict significant damage. Triggers can include events such as login in, placing an order, or registering a credit card.

The graph in Diagram 8 below shows a series of transactions from different IP addresses, with a likely fraud event occurring from IP_1:

IP_x represents distinct IP address, CC_x distinct Credit Card number, ID_x represents the UserID used to carry out the online transaction, and CK_x refers to a specific cookie stored in the system. In this example, one IP has carried out multiple transactions using five credit cards, one of which (CC_1) is used by multiple IDs, where two cookies (CK_1 and CK_2) each share two IDs.

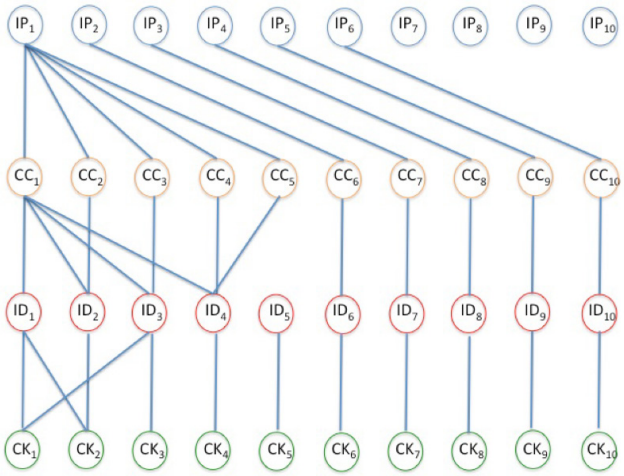

Diagram 8: Online Payment Fraud Originating from Address IP_1

CASE STUDY:

FORTUNE 500 FINANCIAL SERVICES COMPANY

This Fortune 500 financial services company has customers around the world, with online services available in over 30 countries that account for $2.2 million in financial transactions per month.

The company collects a large volume of data—provided by customers and outside vendors—that needs to be analyzed in real time before a transaction can be approved. While the majority of these requests are approved or denied instantly through an automated fraud detection system, potentially fraudulent requests are submitted to an analyst for manual review.

Unfortunately, these manual reviews were taking up to five minutes to perform due to the large number of JOINs required by their Microsoft SQL Server database.

To achieve the real-time results they needed, the team needed a database optimized for storing and traversing several levels of relationships. And with Neo4j, they found exactly what they were looking for.

The development effort with Neo4j was easily completed during spare time. Neo4j provided real-time results with connected data and data visualization that allowed analysts to make faster, more accurate decisions. This opened the door for newer, more extensive searches, which the company hopes to expand from four to 10 degrees of separation.

Conclusion

Whether it is bank fraud, insurance fraud, e-commerce fraud, or another type of fraud, two points are very clear:

The first is the importance of detecting fraud as quickly as possible so that criminals can be stopped before they have an opportunity to do too much damage. As business processes become faster and more automated, the time margins for detecting fraud are becoming narrower and narrower, increasing the call for real-time solutions.

The second is the value of connected analysis. Sophisticated criminals have learned to attack systems where they are weak. Traditional technologies, while still suitable and necessary for certain types of prevention, are not designed to detect elaborate fraud rings. This is where graph databases can add value.

Graph Databases are the ideal enabler for efficient and manageable fraud detection solutions. From fraud rings and collusive groups, to educated criminals operating on their own, graph databases provide a unique ability to uncover a variety of important fraud patterns, in real time. Collusions that were previously hidden become obvious when looking at them with a system designed to manage connected data, using real-time graph queries as a powerful tool for detecting a variety of highly-impactful fraud scenarios.

About the Authors

Gorka Sadowski is Founder and CEO of Akalak, whose mission is to provide Technology and CyberSecurity solutions and services for a better world. Akalak has helped many clients in the US and Europe with their offerings and security posture. More information is available at akalak.com.

Philip Rathle is VP of Products for Neo4j, the world's leading graph database with a history of 24x7 production deployments. Neo4j customers includes a number of Global 2000 organizations spanning a variety of sectors and uses, including fraud detection. To learn more about Neo4j and graph databases, please visit neo4j.com. Additional resources on graph databases are available at graphdatabases.com.

Bibliography and References

1. Experian at http://www.experian.com/assets/decision-analytics/whitepapers/first-partyfraud-wp.pdf
2. Experian at http://www.experian.com/assets/decision-analytics/whitepapers/first-partyfraud-wp.pdf
3. Business Insider 2011 at http://www.businessinsider.com/how-to-usesocial-networks-in-the-fight-against-first-party-fraud-2011-3
4. Gartner at http://www.gartner.com/newsroom/id/1695014
5. Graph Databases, O'Reilly, Ian Robinson, Jim Webber & Emil Eifrem, Chapter 2 (ISBN: 978-1-449-35626-2)
6. Ibid. Pages 5 and 144
7. Coalition against insurance fraud at http://www.insurancefraud.org/article.htm?RecID=3274#.UnWuZ5E7ROA
8. National Insurance Crime Bureau at https://www.nicb.org/newsroom/news-releases/u-s--questionableclaims-report
9. Insurance Fraud Organization at http://www.insurancefraud.org/IFNSdetail.htm?key=17499#.UmmsJyQhZ0o