Paper SAS4368-2020

Network Pattern Match for Identifying Fraud with SAS® Visual Investigator

James Morris, Nicholas Ablitt, SAS Institute Inc.

ABSTRACT

Network analytics plays an increasingly important part in detecting and investigating criminal activity and regulatory compliance issues. SAS® Visual Investigator provides a user interface that, among other things, facilitates the investigation of network activity. These networks are often a combination of relationships formed through common entities referred to in the data (for example, people, locations, and so on) as well as events that tie these entities together. If complex organized activity is detected, a key question an investigator might have is, "Does this activity occur elsewhere in my data?". SAS® provides the NETWORK procedure to analyze graph data, and at SAS® Global Forum 2019, the SAS Cloud Analytic Services (CAS) action patternMatch was introduced, which executes graph queries. Its functionality enables you to search copies of a query graph within a larger graph, with the option of respecting node or link attributes (or both). This paper presents a way for SAS Visual Investigator users to dynamically identify a pattern of interest within their existing investigations, and from that pattern, generate alerts for other occurrences of this activity, thereby putting the ability to answer the key question in the hands of the investigator.

INTRODUCTION

When looking for fraud, it is often not sufficient to look just at an event or individual item of data. Often the relationships between the data unveil additional insights that enable you to determine if something is or isn't fraudulent. A problem often faced when analyzing linked data is determining which of these relationships are interesting and which are not. Network analytics provides us with many ways to analyze the networks formed from this linked data to determine which relationships are most important. SAS® Visual Investigator provides users with an interface for exploring relationships between data, however, in densely linked data, finding the relationships of interest can prove challenging even if an investigator knows what they're looking for. Additionally, if an investigator does find something of interest, they might want to see if that pattern exists elsewhere within the data. SAS® Visual Data Mining and Machine Learning provides the CAS action patternMatch that enables a user to search for a specified pattern across networked data. However, this isn't accessible to users of SAS Visual Investigator. This paper introduces an integration between SAS Visual Investigator and the CAS action patternMatch in the Network action set that enables investigators to search for these relationships of interest themselves. The paper then discusses the different questions that can be answered by this integration, and finally, gives two example use cases to which this integration could be applied.

SAS® VISUAL INVESTIGATOR

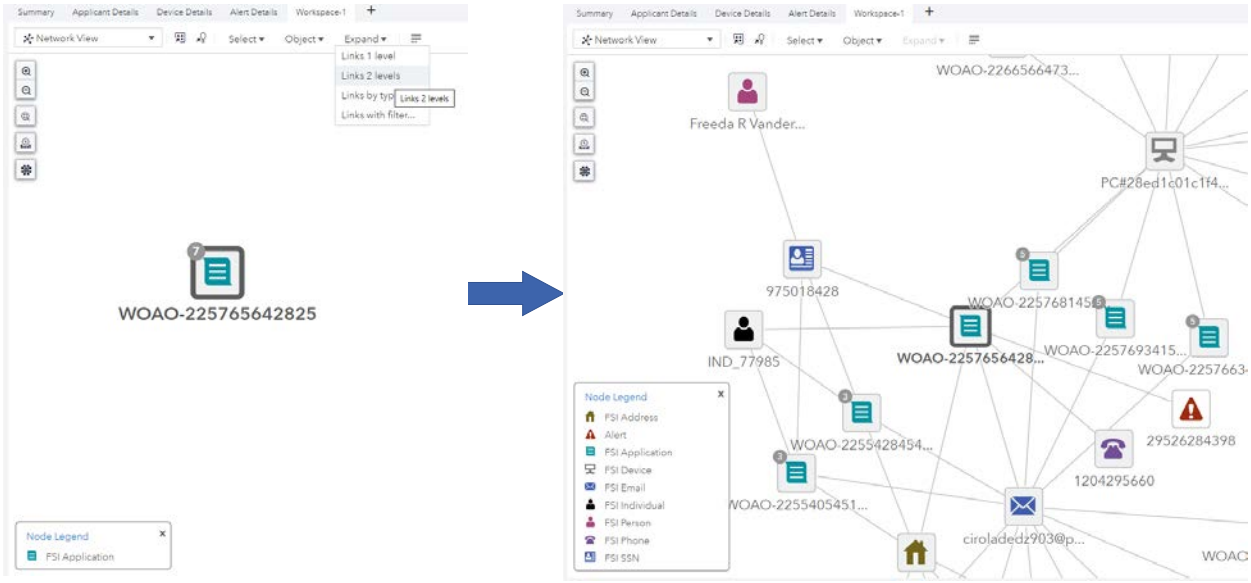
SAS® Visual Investigator is designed to address a wide variety of investigation management and intelligence analysis needs. SAS Visual Investigator has multiple components that help provide investigators with additional insight such as:

- Search and Discovery − Ability to search across data and visualize the data in numerous different ways such as in tabular form, as a network, or in a map

- Alert Triage – Ability to organize and route alerts to different users for investigation and dispositioning
- Case Management – Ability to create cases containing relevant information acquired during an investigation
- Social Network Analysis – Ability to explore the relationships between the data to gain additional insights

Administrators of SAS Visual Investigator can quickly define and alter how the data is modeled and visualized in SAS Visual Investigator by defining entities, relationships between entities, and transactions. The highly customizable user interface (UI) in SAS Visual Investigator can be extended with custom UI components and widgets called Solution Extensions to provide users with additional ways to view and interact with their data.

Entities, relationships, and transactions vary across different business domains. Examples of entities that might be configured for use in detecting loan application fraud at a bank include loan applications, people, addresses. The way the data is modeled and visualized can be thought of as a graph where the entities are the nodes and the relationships and transactions are the links. SAS Visual Investigator provides the users with a way of visualizing and traversing the graph through Network Workspaces. For example, a user of the application fraud detection system might start their investigation by analyzing a suspicious application and then deciding that they wish to know more about the applicant. The user can traverse from the application through the applicant to other applications and beyond to see all related activity and entities as seen in Display 1.



Display 1. Expanding an Application in a Network Workspace within SAS Visual Investigator

Being able to visually explore these relationships can be extremely valuable. However, identifying interesting activity amongst the network of related entities can often be difficult. Here are two questions a user might ask when exploring these relationships:

1. Do the current entities that I am investigating link to anything of interest or form part of an interesting set of relationships? For example, does the loan I am investigating connect to any previously known fraudulent loan applications?
2. Does an observed pattern of interest occur anywhere else in my data? For example, are there other loans in my system that link to any previously known fraudulent loan applications?

Finding patterns in the relationships between entities and attributes is a problem that can be solved using the CAS action patternMatch in the Network action set. In this paper, the patternMatch action is interchangeably referred to as "network pattern match."

NETWORK PATTERN MATCH

SAS® Visual Data Mining and Machine Learning 8.3 on SAS® Viya 3.4 included the new CAS action patternMatch. The patternMatch action is just one of many network analytics algorithms from the Network action set. Others include centrality, community detection, and shortest path, to name a few. The Network action set provides users with a toolkit for analyzing graphs and generating insights that can be used to supplement machine learning models and drive business rules. Graph analytics has applications in many scenarios from modeling the flow of water through pipes to modeling protein-protein interactions in biology. Modeling your system as a graph and applying graph analytics also has many applications within fraud and financial crime. For example, modeling the flow of money through parties can be used to detect money laundering, and understanding the relationships between people and their associated events can uncover organized criminal groups.

A graph is defined as a set of nodes and links. A network is defined as a graph where the nodes and/or links have attributes such as node type, link date. Given a complete graph and a query graph, we are looking to identify the subgraphs of the complete graph that are isomorphic to the query graph. A graph is said to be isomorphic to another if it has the same node and link structure, in other words, if they are topologically identical. The patternMatch action also ensures that the attributes of the query graph have corresponding, matching attributes in the returned subgraphs. In this definition, attributes of the graph refer to one or a combination of these elements: node attributes, link attributes, or a combination of node and link attributes and graph attributes.

Network pattern match enables users to search for many different properties of varying complexity of a graph. Here are some examples:

Topology– By specifying just the links between the nodes, you are able to define the structure of the graph to query for, without specifying any of the attributes of the nodes or links. For example, a user might want to search for the cyclical flow of money but not care much for the attributes of the entities between which the money passes.

Node and Link attributes – By specifying attributes of the nodes and links in the query graph, you are able to perform exact matching on those attributes. For example, you might wish to see all motor insurance claims that are marked as open and that are linked indirectly to a known fraudulent claim.

Inexact Node and Link attributes – Network pattern match enables the user to write functions that evaluate to true or false based on attributes of the nodes or links. These functions enable the user to provide inexact match criteria for node or link attributes. For example, you might wish to filter on motor insurance claims where the claim value is greater than $1000, or filter on people who are marked as either a known fraudster or a suspected fraudster.

Node and Link Pair attributes – Users can also write functions that act on pairs of nodes and links. This method enables the user to specify comparative criteria between nodes and links for matching. An example of this could involve an investigator looking for events that are sequential in time.

Graph attributes – Users can write functions that act across the whole graph. This method enables users to specify match criteria derived from a holistic view of the query graph. This feature could be used to match against aggregate attributes of a subgraph, for example, the sum of all motor insurance claim values on a subgraph. It could also be used to determine whether the average score of all scored entities on a subgraph exceeds a threshold.

It is worth noting that similar functionality is offered by graph database vendors. In the paper "Introducing Pattern Matching for Graph Queries in SAS® Viya® 3.4," the authors provide both a functional and computational comparison of network pattern match with other market leaders such as Neo4j. Significant improvements in run time across a range of standard benchmarks are reported (Galati 2019).

The questions posed in this paper can be answered with network pattern match; however, there might be several factors inhibiting an investigator from performing this activity themselves. First, the interface to network pattern match is through code. Investigators might not have access to or knowledge of how to use a coding environment. Second, an investigator might not know how to use code to prepare the data for use in network pattern match. Third, the investigator may not know how to review the output and relate it to something they have observed within SAS Visual Investigator. Giving investigators the ability to answer such questions is the motivation for the Visual Graph Query Builder, which is a solution extension that integrates the network pattern match capabilities with the graph-like data model found in SAS Visual Investigator. The Visual Graph Query Builder enables users to both perform graph searches and visualize the results in SAS Visual Investigator.

VISUAL GRAPH QUERY BUILDER

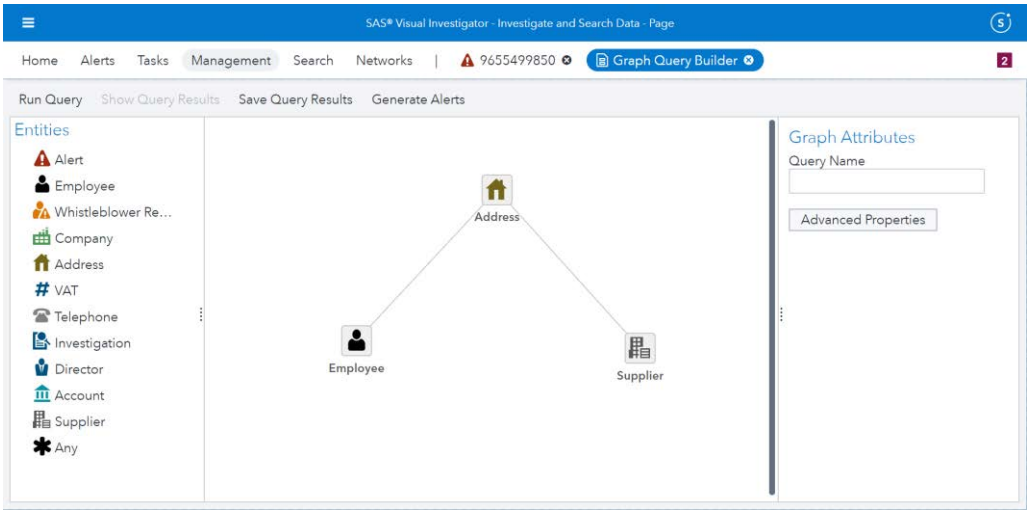
The Visual Graph Query Builder (VGQB) solution extension provides a drag-and-drop user interface for authoring graph queries that are then executed by the pattern match algorithm. To demonstrate how each of the features of pattern match is covered by the solution extension, we will use an instance of SAS Visual Investigator that has been configured for use in identifying anomalies in the procurement cycle. The examples focus mainly on detecting collusion between suppliers and employees.

As discussed above, there are many features of a graph that can be matched using network pattern match. Each of these features is made available in the VGQB and are described in more detail in the following section.

VISUAL GRAPH QUERY BUILDER FEATURES

Relationships between Entities

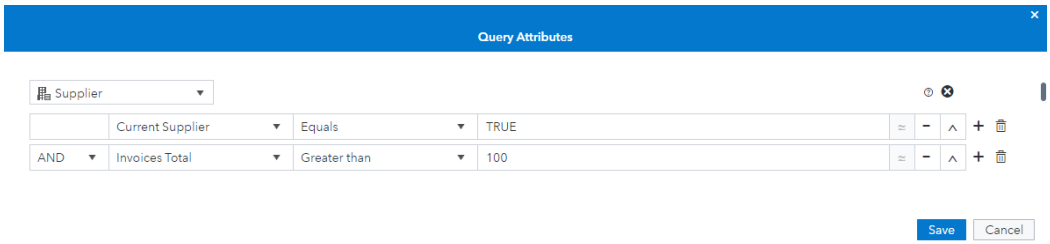
The user can specify a set of entities and relationships to be searched for without specifying any attributes of the entities as seen in Display 2, where the query is looking for any suppliers that share an address with an employee.



Display 2. Query for Employees That Share an Address with a Supplier

Exact and Inexact Entity and Relationship Attributes

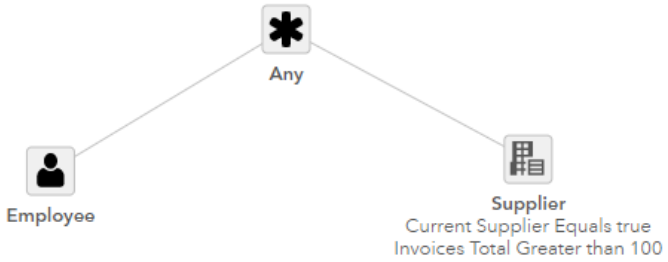
On each of the entities and relationships, the user is able to specify both exact and inexact match criteria as seen in Display 3, where the query is looking for cases where the supplier has more than 100 previous invoices and is a current supplier. This query could also enable users to be more specific about the results they wish to return by restricting the query to a specific instance of an entity. For example, by specifying a supplier's ID, the user can focus on a particular supplier of interest.



Display 3. Visual Graph Query Builder Entity and Relationship Attributes Dialog

Wildcard Entities

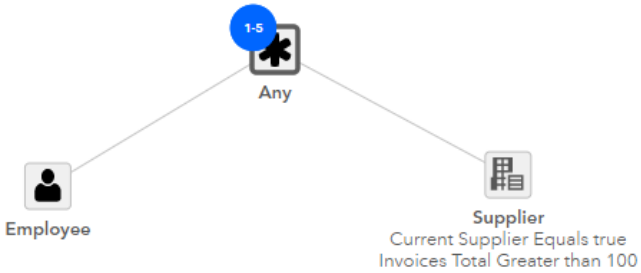
In cases where users do not want to be specific about which type of entity to include in their query, they are able to use a wildcard entity, which can represent any entity. In Display 4, the user is looking for suppliers that are linked to employees through any entity rather than specifically an address.



Display 4. Query from Display 2 and 3 with the Address Entity Replaced with the Any Wildcard Entity

Repeating Entities

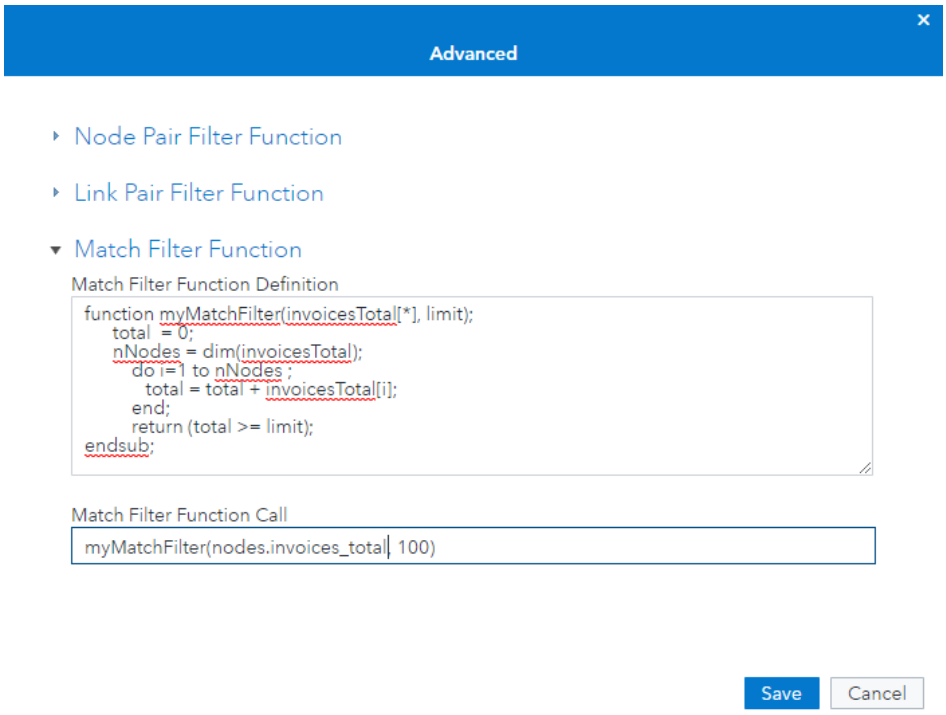
To specify a range of occurrences of an entity, the user is able to configure an upper and lower limit of link traversals. In Display 5, the user is searching for Employees that are linked to a supplier through between one and five wildcard entities.



Display 5. Query from Display 4 Extended So That It Now Returns Suppliers and Employees Indirectly Linked Through between One and Five Entities

Entity Pair, Relationship Pair, and Graph Attributes

The node pair, link pair, and match functions can be added to the query using the Advanced properties dialog as seen in Display 6, where the user has specified that the number of invoices across all suppliers in the subgraph exceeds 100.



Display 6. Visual Graph Query Builder Advanced Properties Dialog Containing a Match Filter Function

These features provide a range of functionality that enables you to construct complex queries. To enable an investigator to make best use of this, the integration between SAS Visual Investigator and network pattern match also requires some thought about the process that the investigator would follow through the system. The investigator must consider where the pattern comes from, how it gets reviewed and refined, and what happens with the results.

PROCESS FLOWS

Earlier we mentioned two use cases:

- Does the object I'm currently investigating link to anything suspicious?
- Does this suspicious pattern occur anywhere else in the data?

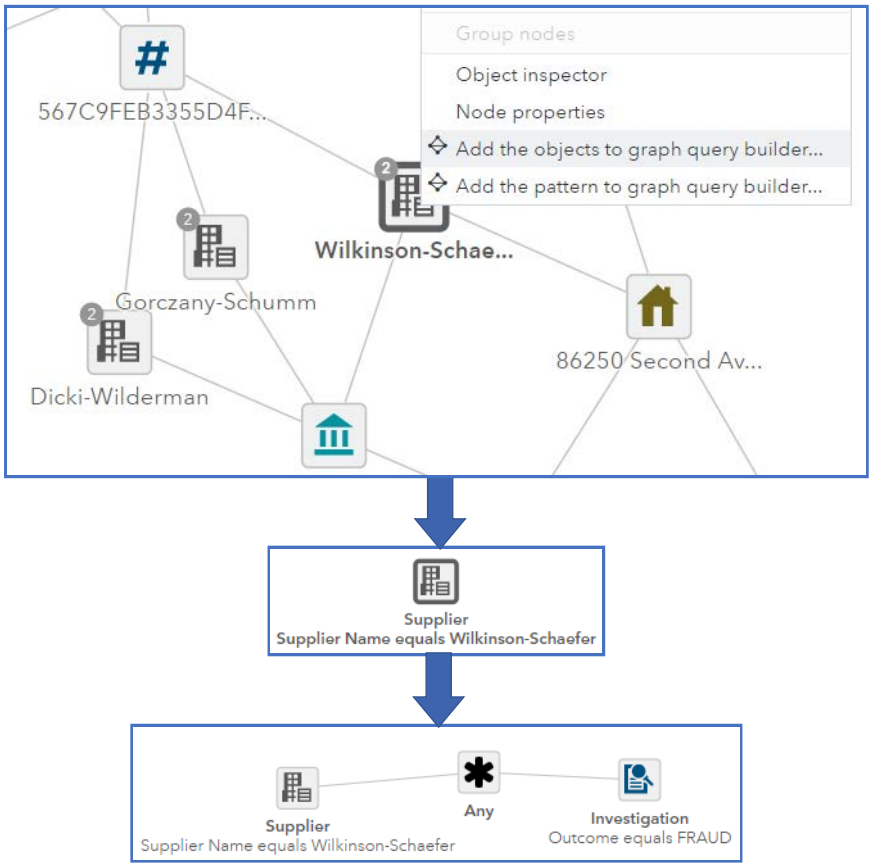
Below we discuss how the VQGB enables the user to deal with these use cases, whether you are starting from an idea or a current investigation.

Investigation of Specific Instances of Entities

The starting point in this use case is specifying a particular entity or set of entities. The entity might be currently under investigation such as a supplier as seen in Display 7, or it might be just an entity known to the investigator. Without the VGQB, if the user wanted to know whether a supplier under investigation is linked to any known fraudulent suppliers, they would have to recursively expand through each node in every direction and analyze all the entities and relationships brought in with each expansion to see if the pattern of interest exists. The process of observing whether a specific pattern occurs might also involve

clicking through each entity and relationship to observe each of their attribute values. In densely linked data, this quickly becomes unmanageable as with every layer of depth, the number of entities and relationships in the Network workspace increases significantly.

With the VGQB, the user can select specific instances of entities they wish to seed the VGQB with, and then use the drag-and-drop interface to enhance the query. Display 7 contains an example showing a query starting with just the selected supplier and then looking for links to previously confirmed fraudulent suppliers. It's worth noting that in moving from the Network workspace to the VGQB, relevant attributes of the selected entities are brought into the VGQB and used as a starting point for the query. This can be seen in Display 7 where the supplier name, Wilkinson-Schaefer, is included in the query. It could also be that a user starts directly from the VGQB and builds out the query from scratch based on an idea rather than starting the query building process from specific entities within an investigation.



Display 7. Selecting a Specific Entity from the Network Workspace, Using It as a Starting Point in the Visual Graph Query Builder, and Then Adding Additional Entities to the Query

Find Other Examples of a Pattern of Interest

As an investigator, you might be looking at a workspace within SAS Visual Investigator and discover a set of relationships that indicates activity that is of interest. You might carry out your investigation, but as a follow-on want to look further into the pattern of activity behind it. The question being asked is whether this activity is more widespread than the specific example that has been found.

Typically, this has been a difficult question to answer. For a specific example, the interface (in this case, SAS Visual Investigator) that the user must interrogate is well suited to its task, but poor at answering this more general question. A few options are open to an investigator:

- to look for characteristics of individual entities and search for those in the hope that they also display this networked behavior.
- to pass on the details of what has been found to an analyst team who have additional capability available to them, such as the SAS Network action set.

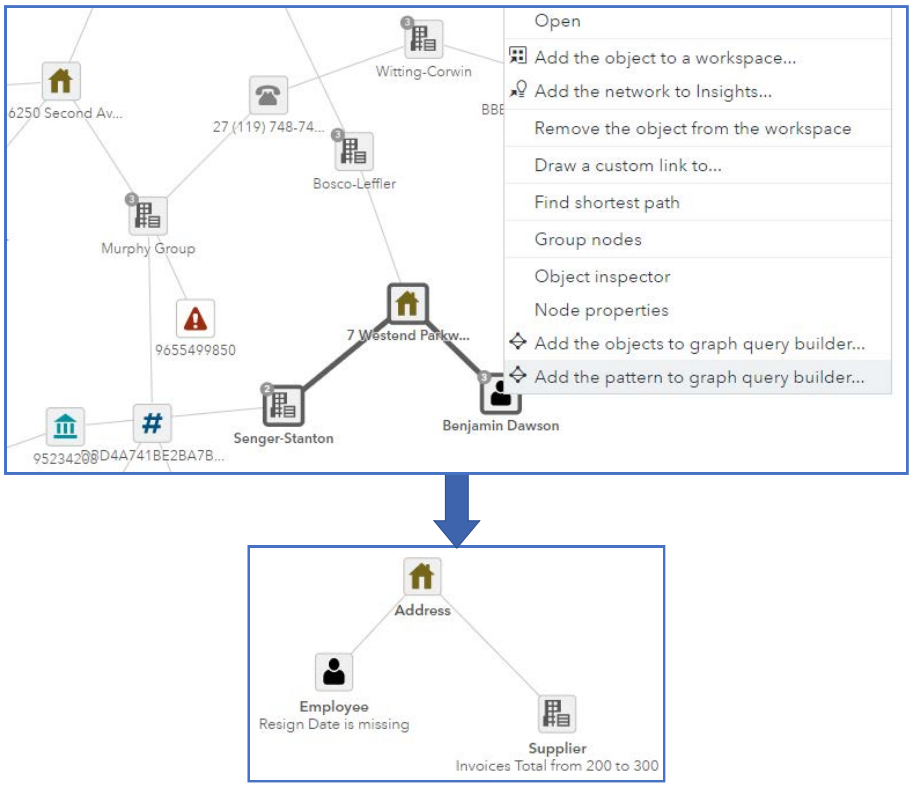
Neither of these options is ideal. The first approach, if it works at all, will find only very specific examples and doesn't recognize the contribution of the organized activity that might be happening between the different parties involved. The second approach might provide the required results, but it suffers from the disconnect of the investigators and the analysts.

A description of where and how these disconnects can happen in the process:

1. The investigator has only a specific example. From that example, the investigator has to understand what is the generalized activity of interest and what is specific just to this example. For example, the employee providing the same address as a supplier is interesting, as is the fact that they have only recently become a supplier. The fact that there has been seven invoices for a total of $150,000 is interesting, but isn't essential.
2. This must be communicated to the analysts in a way that they can interpret and understand. Is this a written description? Does it use an agreed-upon grammar? Is there an export of the example from the investigation tool? The analysts might be familiar with field names in the data but not the labels on items in the investigation tool, so the analyst has to interpret the investigators' request.
3. The analyst must then make decisions on how they can look for this across the full set of data. They require access to a tool such as the SAS Network action set and the ability to turn their interpretation of the investigators' request into an appropriate data query.
4. The analyst must be able to interpret the results to understand if the results also demonstrate valid and interesting examples. Typical coding environments are not well suited to visualizing and interrogating graph data. If the analyst has an appropriate tool available to them, they also then need to understand whether the examples they have found are of interest to the investigator. To determine this, the results must be passed back to the investigator. At this point, the investigator might choose to refine their request and iterate through the above process.

As can be seen, the process can be very inefficient, require multiple people, and take a significant amount of time to iterate through to a set of results that are of interest. Answering the question using this process is hard.

A better solution to this problem is to provide the investigator with the ability to go through this cycle themselves. With this solution extension, once an investigator has found a pattern they wish to search for, they can seed the Visual Graph Query Builder with the entities and relationships that make up the interesting pattern, this can be seen in Display 8, where an interesting pattern has been observed while exploring the network of an alerted supplier and is used in the query builder. Whether a pattern is interesting may not depend entirely on the topology, the entity and relationship attributes can also determine whether a pattern is worth querying for. When seeding the query builder with a set of entities and relationships from a network, relevant attributes are brought through as a starting point for the query.



Display 8. Selecting a Specific Pattern in a Network Workspace and Using It in the Visual Graph Query Builder

When using this approach, the initial pattern might be quite specific and contain no fuzziness. The user might then be able to abstract the question a bit more. Using the example from Display 8, the user might conclude that what they're actually interested in is any relationship between a supplier and an employee. As seen in Display 9, the initial query might then be made more generic and fuzzier by using a less restrictive search or a more generally defined depth at which the user can expect to find the pattern.

Display 9. Fuzzy Query for Showing Employees That Are Indirectly Linked to Suppliers Through between One and Five Wildcard Any Entities

From the results of this less exact query, a user might, after investigation of the results, decide that there are many distinct patterns that are uncovered by the initial query. From this point, the user might decide to create a series of queries for each of the distinct pattern types like the one seen in Display 10, where the user is looking for active suppliers that are indirectly linked to an employee through being a subsidiary of a company that has a director that shares an address with the employee. The results of each of these distinct pattern types might be treated differently depending on the how interesting the pattern is believed to be. Dealing with the output of these queries is discussed further in the Output section.

Display 10. Visual Graph Query Builder Containing a More Specific Pattern

It is worth noting that you do not have to observe a pattern in the data to query for it. If the user has an idea for a pattern of interest they want to search for, they can build out the pattern from scratch without going via a Network Workspace.

Output

Once a query has been defined through either of the above two processes, the user will next want to execute the query and visualize the results. By running the query, the solution extension generates a list of each of the matches. From this list, a user is able to select one or many of the matches to add to a new or existing workspace. After inspection of the matches produced by the query, the user can go back into the query builder, make adjustments to the query, and run the new query. This process can be iterated on until the user is satisfied that they have found something of interest. At that point, they can choose to do any of the following tasks:

- Add the pattern to an existing investigation's workspace - In the case where the investigator wants to know if particular entities are part of an interesting pattern, show and highlight the pattern in the workspace of an investigation.
- Add the pattern to a new investigation - Start an investigation based on the occurrence of a pattern.
- Use the pattern in alert generation - Write business rules on the occurrence of a pattern or using the existence of the pattern as a feature of an entity that is used in a machine learning model.
- Use the pattern as an attribute of an entity - Save the pattern as an attribute of an entity and allow the user to visualize the pattern(s) when looking at an entity. For example, when investigating a particular supplier, indicate that the supplier is present on some interesting patterns and allow the user to visualize those patterns.
- Save the pattern as an entity - Patterns could also be modeled as entities within SAS Visual Investigator, enabling them to be scored, alerted on, or investigated as with other entities within SAS Visual Investigator.

USE CASES

This section presents two different domains to which this technology could be applied. There are countless other examples of where this technology could prove beneficial, but here we will focus on just procurement and missing trader intra-community fraud.

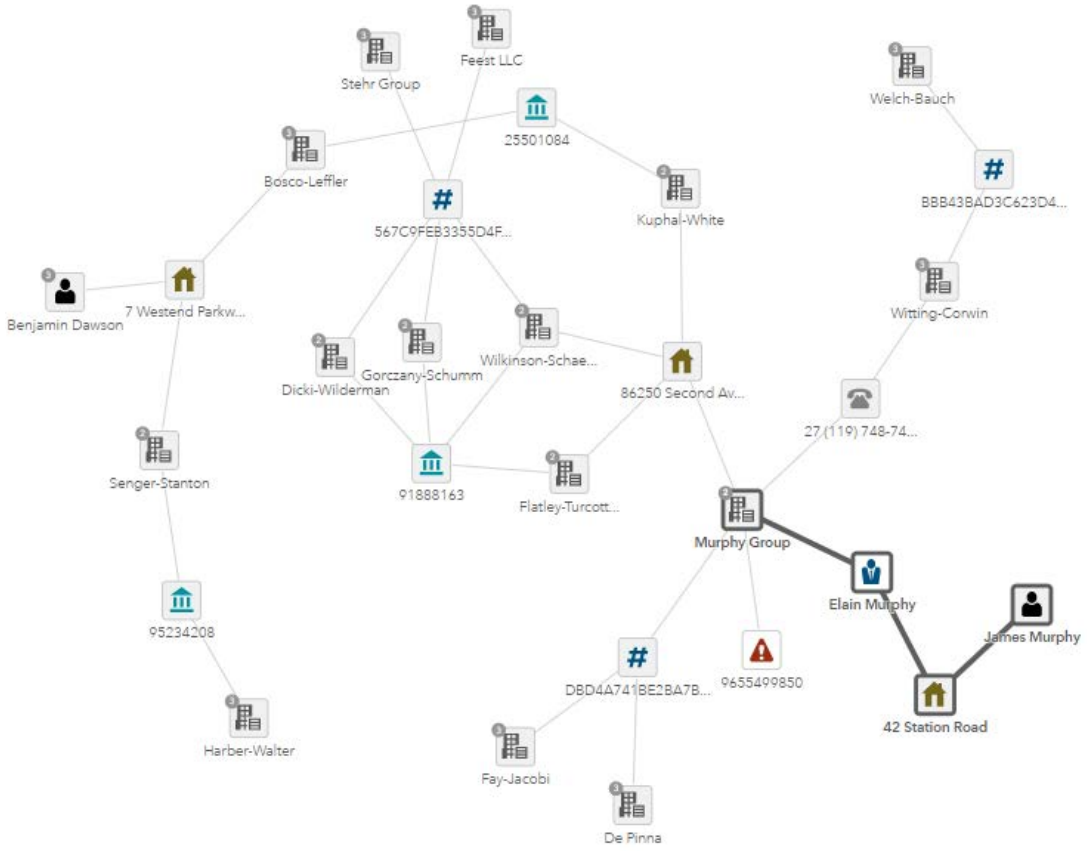
PROCUREMENT INTEGRITY

Procurement is the process of acquiring goods or services on behalf of an organization. Typically, when procuring goods or services, organizations go through a procurement cycle as illustrated in Figure 1. Any attempt to influence with one of the steps in the cycle for personal financial gain is of concern to the integrity of the procurement process.

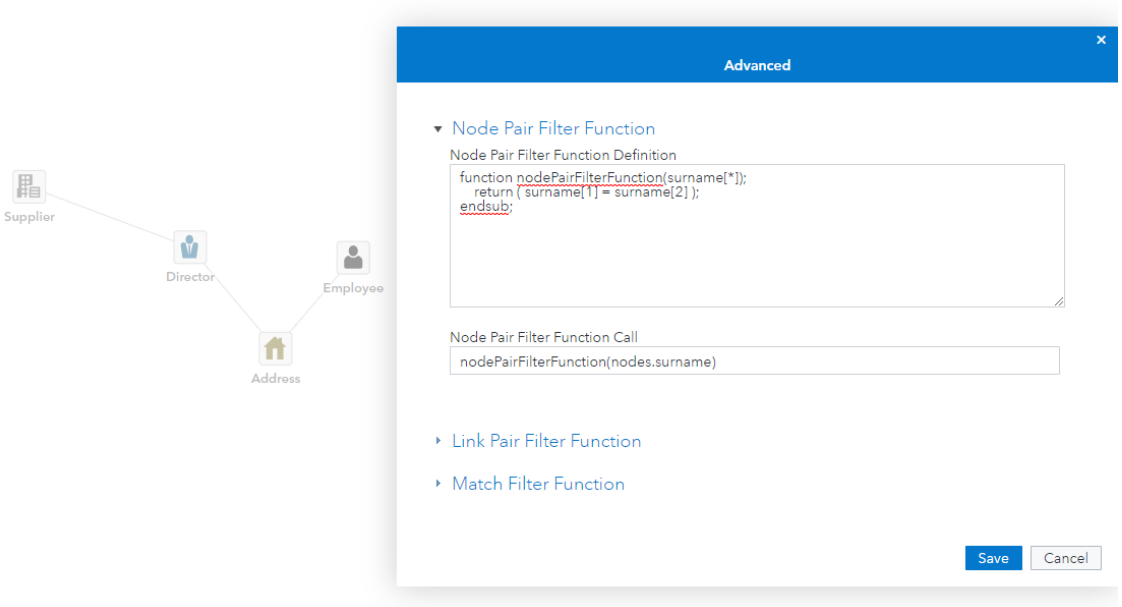
Figure 1. Graphic Depiction of the Steps in the Procurement Cycle (See References)

Continuous monitoring within procurement covers a wide array of different activities. One example involves kickbacks where an employee in charge of procuring a particular good or service receives some form of payment from the vendor for choosing that vendor. Another example is where a supplier might try to submit duplicate invoices in the hope they will be paid without scrutiny.

Although a lot of detection in procurement systems is done by assessing single events such as invoices and purchase orders, finding conspiring vendors and employees also plays a large part in detecting malpractice and is a good use case for using network pattern match and the Visual Graph Query Builder. In typical procurement monitoring systems, there are rules in place that search for employees or suppliers that share addresses, telephone numbers, or bank accounts with suppliers. However, on investigation of a suspicious supplier, an investigator might discover an unfamiliar indirect link between the supplier under investigation and an employee. This indirect link might lead them to ask the question: Does this pattern occur anywhere else in the system? This question sends them down the path described in the section Find Other Examples of a Pattern of Interest. For example, imagine that the investigator noticed an employee sharing an address and having the same surname as a director of an existing supplier as seen in Display 13. This pattern can be used as a starting point in the VGQB as seen in Display 14.



Display 13. Suspicious Pattern Identified While Investigating the Network Workspace of an Alerted Entity



Display 14. Query for Pattern Observed in Display 13

The query from Display 14 contains the pattern from Display 13 and is looking for suppliers where the director shares an address with an employee and the director and employee have the same surname. An earlier example in Display 8 – Display 10

presented another pattern that was noticed whilst exploring a Network Workspace which was the catalyst for the user asking a much broader question about how suppliers and employees can be related. Ultimately this broad pattern was refined to a much more specific instance of a pattern that captured a specific relationship between suppliers and employees.

If these patterns are believed to be indicative of employees influencing and benefitting from the choice to procure goods from that supplier, these patterns can be added to the set of patterns used in an alert generation process.

MISSING TRADER INTRA-COMMUNITY FRAUD

A missing trader intra-community (MTIC) or value-added tax carousel fraud detection system is another good use case for pattern match and the VGQB. MTIC fraud occurs when organizations abuse cross-border trading laws to receive unlawful VAT repayments from governments. An simplified example that highlights the steps in MTIC fraud is given in Figure 2.

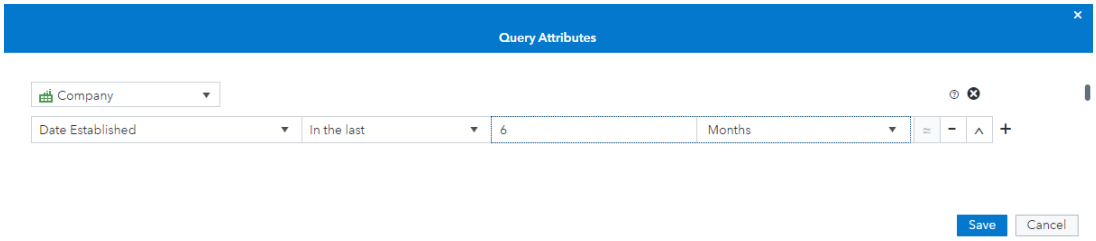
Figure 2. Illustration of the Steps That Take Place in VAT Carousel Fraud (See References)

VAT Carousel and MTIC fraud are known issues, but they are often very difficult to identify in a network of many companies, each with multiple different trades between one another, especially when the companies committing such crimes are deliberately trying to obfuscate their fraudulent activity. You might ask the question, is the organization that I'm currently investigating, in this case Johnson and Cooper Limited, part of any pattern that might be indicative of MTIC or VAT Carousel Fraud. In this example, a user would follow the process flow laid out in section Investigation of Specific Instances of Entities.

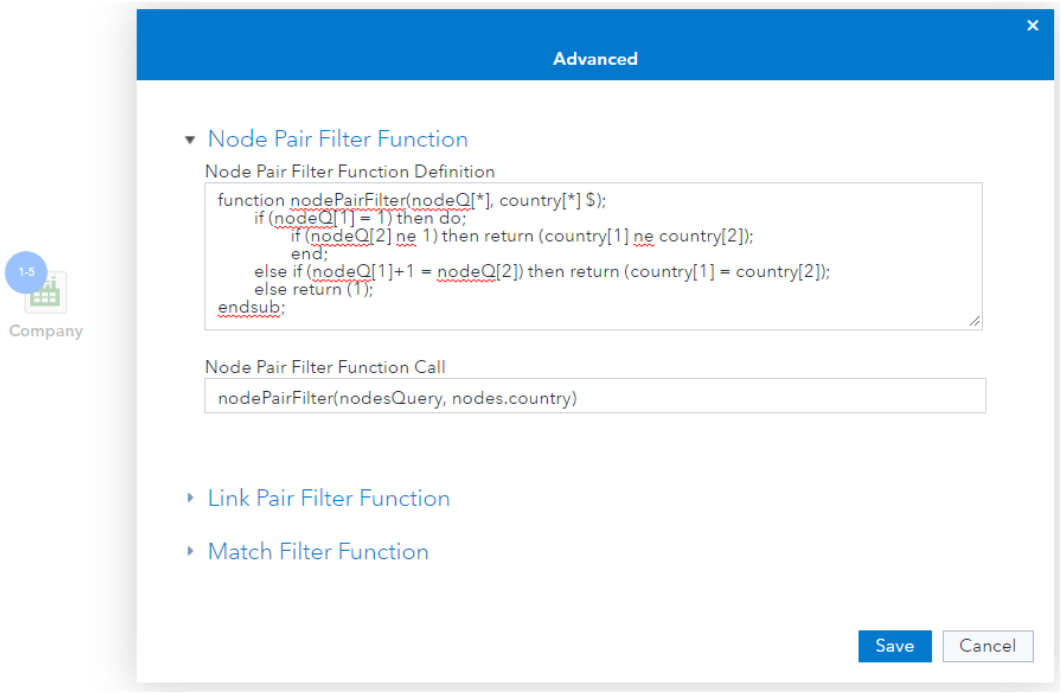
An example of the query that might be built to detect VAT Carousel fraud is shown in Display 15 – Display 18. Here, the user is looking for the following patterns:

- The company under investigation is linked to between one and five other companies.
- Goods are traded from a company in one EU country to a company in another EU country and then back to the originating company.
- Companies in the cycle are all recently established.
- All trades are sequential in time.
- All trades happen within a short period of time.

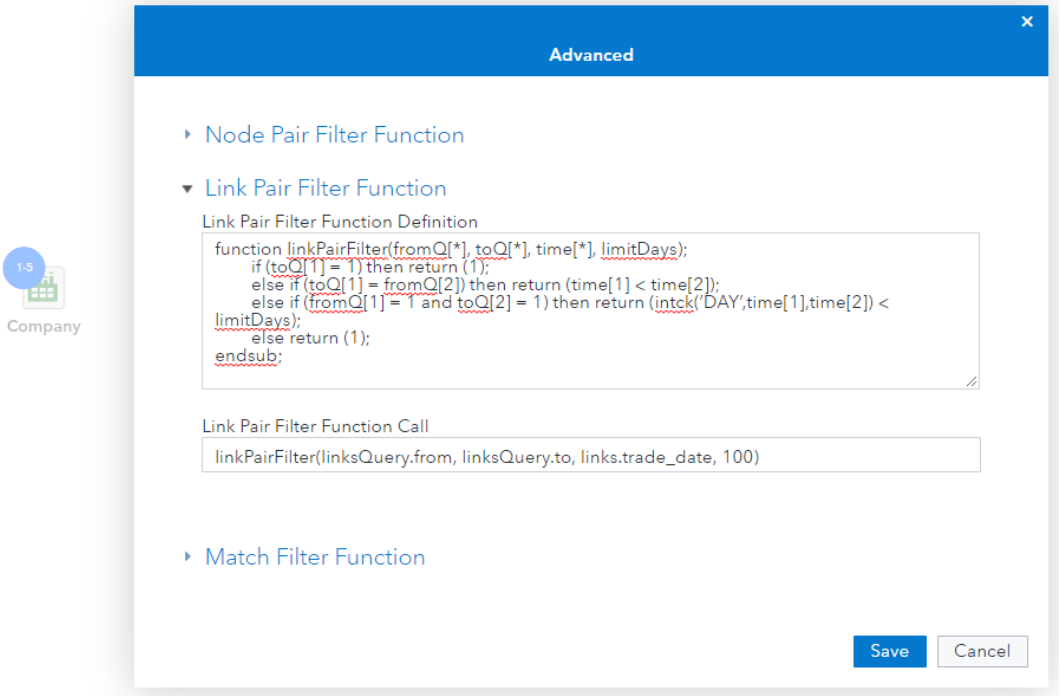
Display 15. Query for a Specific Company Linked to between One and Five Other Companies



Display 16. Query Attributes Dialog Specifying That the Date Established on Each of the Companies Is Within the Last 6 Months



Display 17. Node Pair Filter Function to Detect Where the Sequence of Events Starts and Ends in the Same Country All Other Trades Occur between Countries Registered in a Second Country



Display 18. Link Pair Filter Function to Make Sure All Trades Are Sequential in Time and That They Happen within 100 days of One Another

With the results from this query, the user could add and highlight any matches to the Network Workspace of the investigation of Johnson and Cooper Limited, thus revealing any suspected VAT Carousel fraud rings that this company is a part of. In addition, they might decide a more generic version of this query, that doesn't only focus on Johnson and Cooper Limited, is worth including in an alert generation process.

CONCLUSION

By creating a solution extension that integrates the inherent graph data model within SAS Visual Investigator, with the range of capabilities offered by network pattern match, users of SAS Visual Investigator can quickly and easily answer important questions about their data. The Visual Graph Query Builder solution extension enables the investigator to identify more complex indicators of suspicious activity that include not only information about a single entity that could be uncovered using the SAS Visual Investigator native search, but also attributes of linked entities and the relationships between them.

For an investigator, manually looking for complex relationships extending from an investigation is difficult and time consuming. Broadening that to look for all cases of the behaviour would be impossible for an investigator with their current capabilities. However, the Visual Graph Query Builder gives the investigator access to information that would otherwise require the investigator to work with an analyst across different platforms. The results enable investigators to both enhance their investigations and enrich an alert generation process.

REFERENCES

Belastingdienst. "Btw-carrouselfraude." https://www.belastingdienst.nl/wps/wcm/connect/bldcontentnl/belastingdienst/zakelijk/btw/btw_aangifte_doen_en_betalen/btw-fraude/btw_carrouselfraude/btw_carrouselfraude. (Access the figure in English at https://en.wikipedia.org/wiki/Missing_trader_fraud.)

Blue Ocean Training & Consultancy. "Procurement Courses." ("Procurement Cycle" image.) http://www.blueoceanacademy.com/procurement-courses/

Galati, Matthew, Steve Harenberg, Yi Liao, and Brandon Reese. 2019. *SAS® Visual Data Mining and Machine Learning 8.5: The NETWORK Procedure*. Cary, NC: SAS Institute Inc. https://documentation.sas.com/?docsetId=casmlnetwork&docsetTarget=titlepage.htm&docsetVersion=8.5&locale=en.

Galati, Matthew, Steve Harenberg, and Rob Pratt. 2019. "Introducing Pattern Matching for Graph Queries in SAS® Viya® 3.4." *Proceedings of the SAS Global Forum 2019 Conference.* Cary, NC: SAS Institute Inc. https://www.sas.com/content/dam/SAS/support/en/sas-global-forum-proceedings/2019/3353-2019.pdf.

RESOURCES

Europol. 2020. "MTIC (Missing Trader Intra-Community) Fraud." Accessed January 10, 2020. https://www.europol.europa.eu/crime-areas-and-trends/crime-areas/economic-crime/mtic-missing-trader-intra-community-fraud

Financier Worldwide. 2017. "Procurement fraud – an old fraud flourishing in emerging markets and costing businesses billions." Accessed February 01, 2020. https://www.financierworldwide.com/procurement-fraud-an-old-fraud-flourishing-in-emerging-markets-and-costing-businesses-billions#.XjcOd8j7SHs

US Department of Defense. 2018. "Semiannual Report to the Congress." Accessed February 10, 2020. https://media.defense.gov/2018/Dec/06/2002069859/-1/-1/1/SAR%20SEPT%202018.PDF.

ACKNOWLEDGMENTS

I would like to thank Manoj Chari for his role in envisioning the integration between network pattern match and Visual Investigator. I would also like to thank Matthew Galati, Brandon Reese, and Stephen McKenna for all their support with combining the two technologies, and finally Nick Feast for helping define the use case.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

James Morris
SAS Institute Inc.
7th Floor 199 Bishopsgate,
London, EC2M 3TY, United Kingdom
James.Morris@sas.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.