ORACLE

17 Use Cases for Graph Databases and Graph Analytics

Table of Contents

Introduction

Let's say you have to perform social network analysis, uncover fraudulent bank transactions, or provide product recommendations.

Often, discovering the answer to each of these questions can be complicated and possibly time consuming too.

But with graph database, you can view the data landscape in a completely new way. Discover new insights. Solve complex problems. Unlock endless possibilities.

Why are graphs important?

Graph technologies have become a groundbreaking way for organizations everywhere to address uses that other methods simply can't address in an efficient manner. In fact, for two years running, Gartner selected graphs as one of their top analytics and data trends because of the significant potential for disruption. In today's world, companies know that they must be innovative—or be disrupted.

Graphs capture relationships and connections between data entities. Those relationships and connections can be used in data analysis. Much of data is connected, and graphs are becoming increasingly important because they make it easier to explore those connections and draw new conclusions.

Graphs and graph databases provide graph models to represent relationships. They allow users to apply pattern recognition, classification, statistical analysis, and machine learning to these models, which enables more efficient analysis at scale against massive amounts of data.

When it comes to analyzing graphs, algorithms explore the paths and distance between the vertices, the importance of the vertices, and clustering of the vertices. The algorithms will often look at incoming edges, importance of neighboring vertices, and other indicators to help determine importance.

Because graph databases explicitly store the relationships, queries and algorithms utilizing the connectivity between vertices can be run in sub-seconds rather than hours or days. Users don't need to execute countless join and the data can more easily be used for analysis and machine learning to discover more about the world around us.

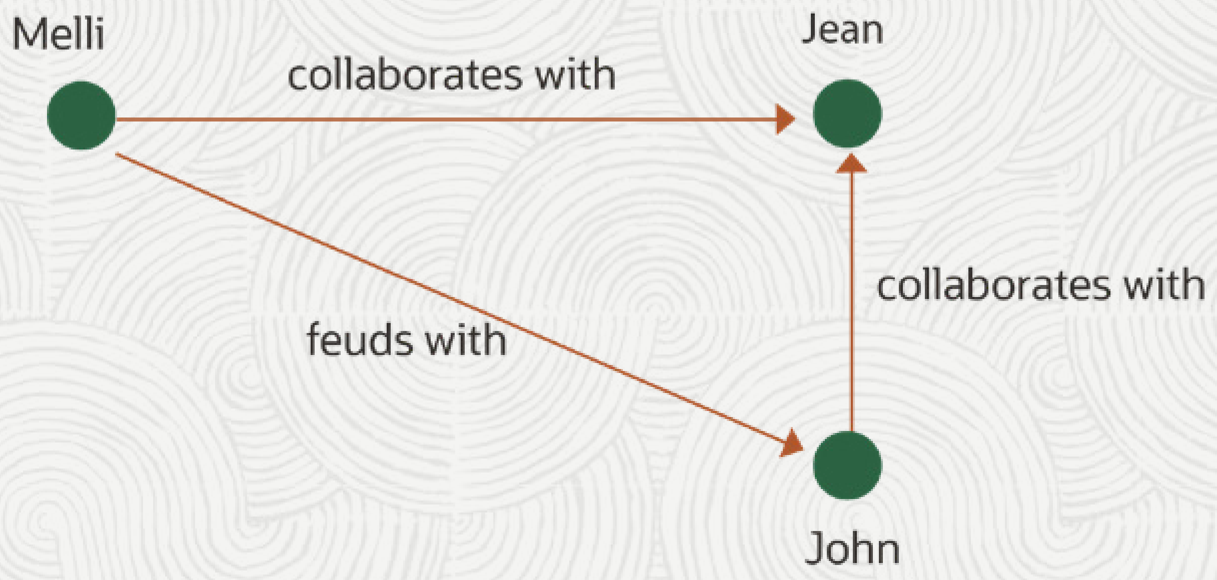
What is graph technology?

There are two types of graphs: property graphs and RDF graphs. The property graph focuses on analytics and querying, while the RDF graph emphasizes data integration. Both types of graph consist of a collection of points (vertices) and the connections between those points (edges). But there are differences as well.

Property graphs are used to model relationships between data, and they enable query and data analytics based on these relationships. A property graph has vertices that can contain detailed information about a subject, and edges that denote the relationship between the vertices.

In the example below, Melli, Jean, and John are all vertices and "collaborates with" and "feuds with" are the edges denoting the relationships between each vertex. Each vertex can contain more information about Melli, Jean, and John, such as where they live and what they like.

Because they are so versatile, property graphs are being used in a broad range of industries and sectors, such as finance, manufacturing, public safety, retail, and many others.



RDF graphs (RDF stands for Resource Description Framework) are designed to represent statements and are best for representing complex metadata and master data. They are often used to represent complex concepts in a domain, or in situations that require rich semantics and inferences on data.

In the RDF model a statement is represented by three elements: two vertices connected by an edge. Every vertex and edge is identified by a unique URI, or Unique Resource Identifier. The RDF model provides a way to publish data in a standard format with well-defined semantics, enabling information exchange. Government statistics agencies, pharmaceutical companies, and healthcare companies are among the types of organizations that have adopted RDF graph.

17 property graph use cases

Organizations everywhere are turning to graph technology. In this ebook, we'll walk you through a few of the most popular uses of graph, organized across the following industries and categories:

- Financial services
- Manufacturing
- Government
- Data regulation and privacy
- Marketing
- AI and machine learning research

Financial services

No matter how hard they try, financial criminals are linked by relationships—whether it's relationships to other criminals, locations, or of course, bank accounts. Graph technology takes advantage of this fact to unfold new possibilities in the financial services world.

Money laundering

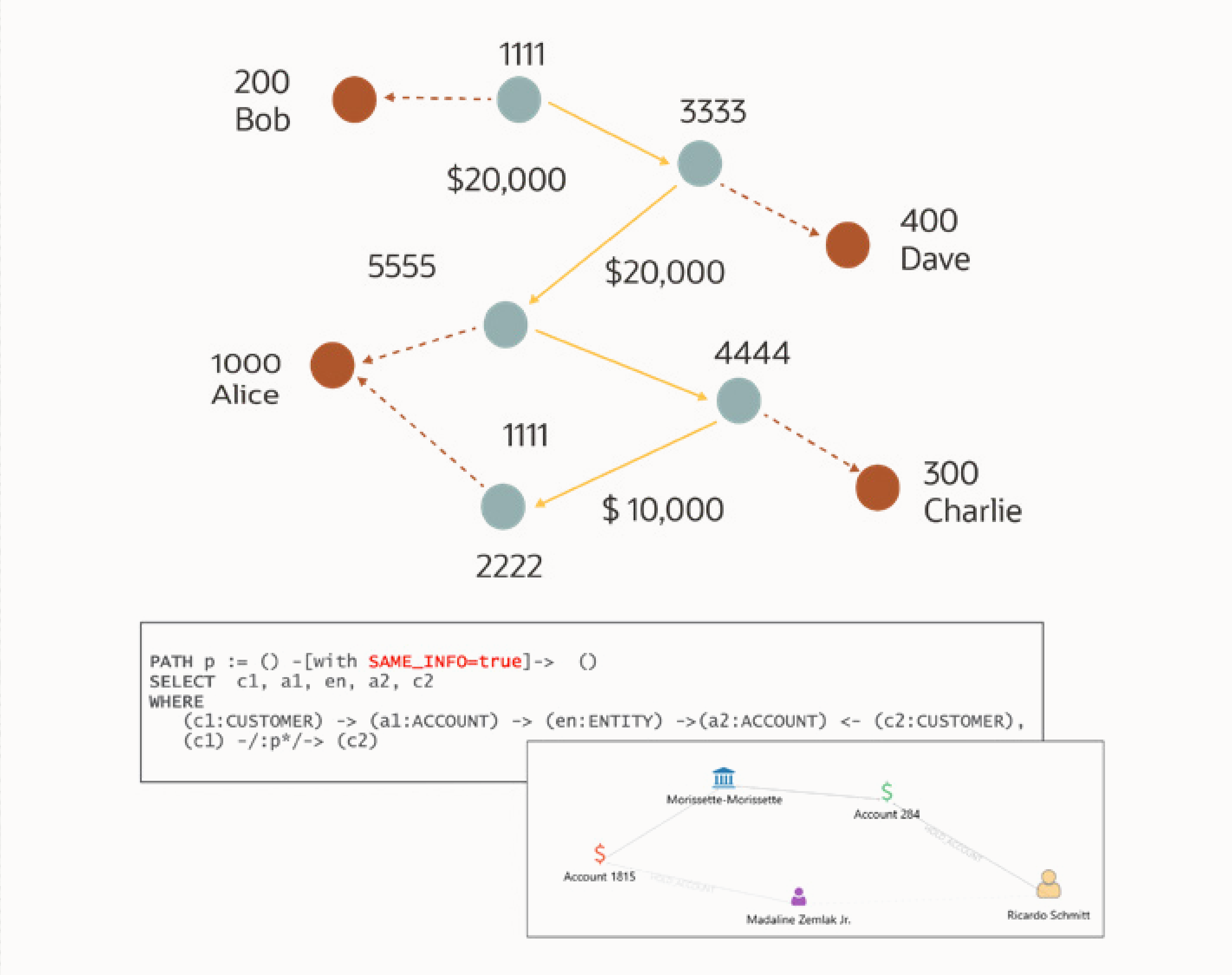
The problem

Conceptually, money laundering is simple. Dirty money is passed around to blend it with legitimate funds and then turned into hard assets. This is the kind of process that was used in the Panama Papers analysis.

More specifically, a circular money transfer involves a criminal who sends large amounts of fraudulently obtained money to himself or herself—but hides it through a long and complex series of valid transfers between "normal" accounts. These "normal" accounts are actually accounts created with synthetic identities. They typically share certain similar information because they are generated from stolen identities (email addresses, addresses, etc.) and it's this related information that makes graph analysis such a good fit to make them reveal their fraudulent origins.

The graph solution

To make fraud detection simpler, users can create a graph from transactions between entities as well as entities that share some information, including the email addresses, passwords, addresses, and more. Once a graph is created, running a simple query will find all customers with accounts who have similar information, and reveal which accounts are sending money to each other.



```
PATH p := () -[with SAME_INFO=true]->  ()
SELECT  c1, a1, en, a2, c2
WHERE
   (c1:CUSTOMER) -> (a1:ACCOUNT) -> (en:ENTITY) ->(a2:ACCOUNT) <- (c2:CUSTOMER),
   (c1) -/:p*/-> (c2)
```

Detecting money mules and mule fraud

The problem

Mule fraud involves a person, called a money mule, who transfers illicit goods. This can involve drugs but when it comes to the financial industry, usually involves money. The money mule transfers money to his or her own account, and the money is then transferred to another scam operator who is usually in another country.

Traditionally, rule-based models create alerts and the suspicious accounts are flagged by humans. Machine learning is also used to predict human decisions.

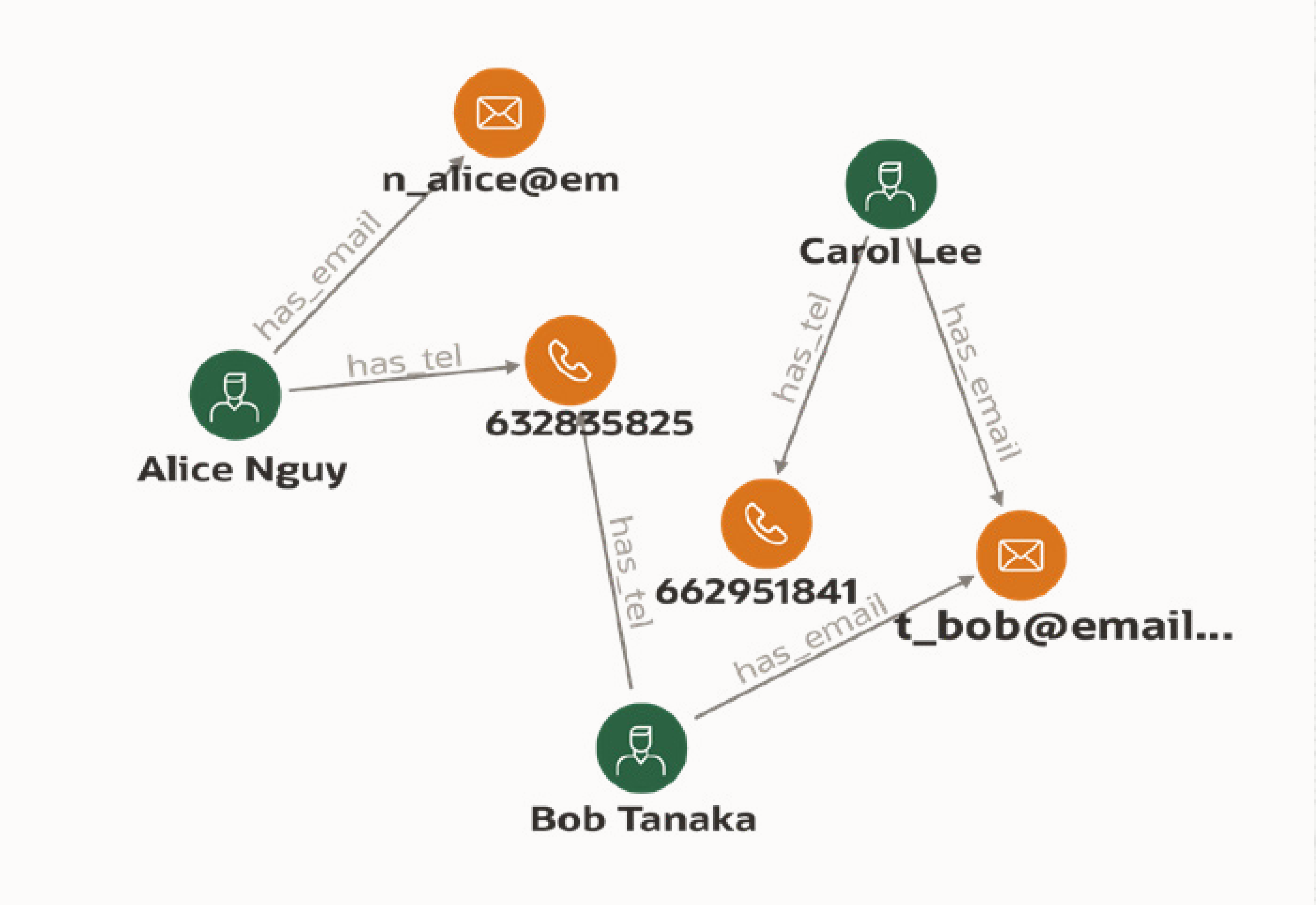
However, it is often difficult to improve the models because the accounts themselves usually have limited information.

The graph solution

This is where graphs come in. With graph technology, users can take the transaction information as edges and generate more features of the accounts based on surrounding relationships and transactions. For example, by using graph-based centrality scores, users can determine how close certain accounts are to known mule accounts.

In addition, these false accounts often share similar information (such as address or telephone numbers) because such information is necessary for registering the accounts—and the criminals only have so many identities to draw from. By using graph-based queries, graph users can quickly discover the accounts with similar relationships or the accounts involved with patterns like circulation and flag them for further investigation.

Through this method, graph technology can enhance machine learning models trained to discover money mules and mule fraud.



Real-time fraud detection

The problem

In today's world, consumers demand instant access to services and to money transfers—which opens up opportunities to criminals. For example, payment services apps try to deliver money as quickly as possible to valid users while also ensuring money isn't sent for illicit purposes or hiding the real receiver by getting sent in circuitous routes. This necessitates real-time fraud detection.

The graph solution

Because graphs enable lightning-fast answers to queries and because they expand access to data, they have become a popular technology in the realm of real-time fraud detection. When investigating transactions with graph technology, it's not only the transactions that can be modeled in graphs. Graphs are extremely flexible, which means the heterogeneous surrounding information can also be modeled. For example, client IP addresses, ATM geolocation, card numbers, and account IDs can all become vertices, and the connections can all become edges.

Property graph is often used for fraud detection, especially in online banking and ATM location analysis because users can design the rules for detecting fraud based on datasets. For example, detection rules can be set up for:

- IPs which log in with multiple cards registered in different places
- Cards used in different places with very far distances
- Accounts receiving one-time inbound transactions from other accounts registered in various places

These rules can be applied real-time because Oracle's graph technologies can:

- Keep graphs updated and synchronized to the original relational table dataset
- Run high-performance queries and algorithms

Manufacturing

Manufacturing is all about relationships and dependencies, which makes graph technologies a perfect fit for discovering more information in a speedy manner.

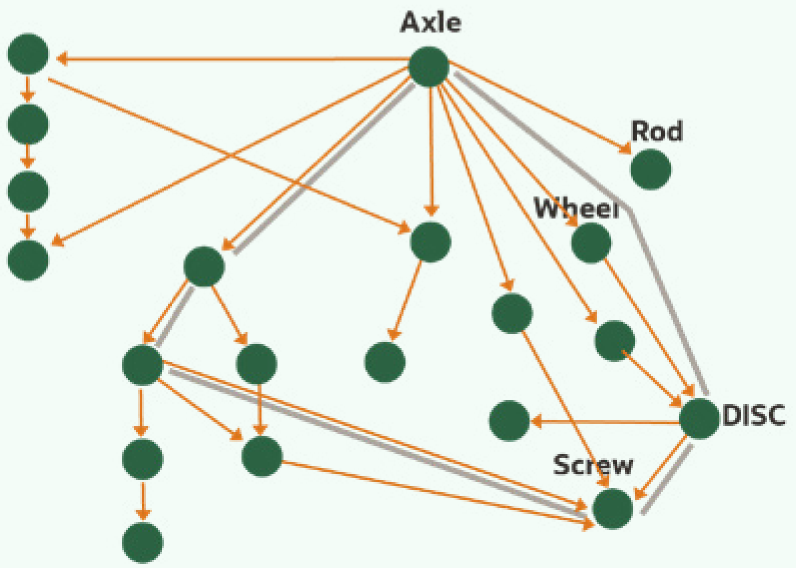
Bill of materials

The problem

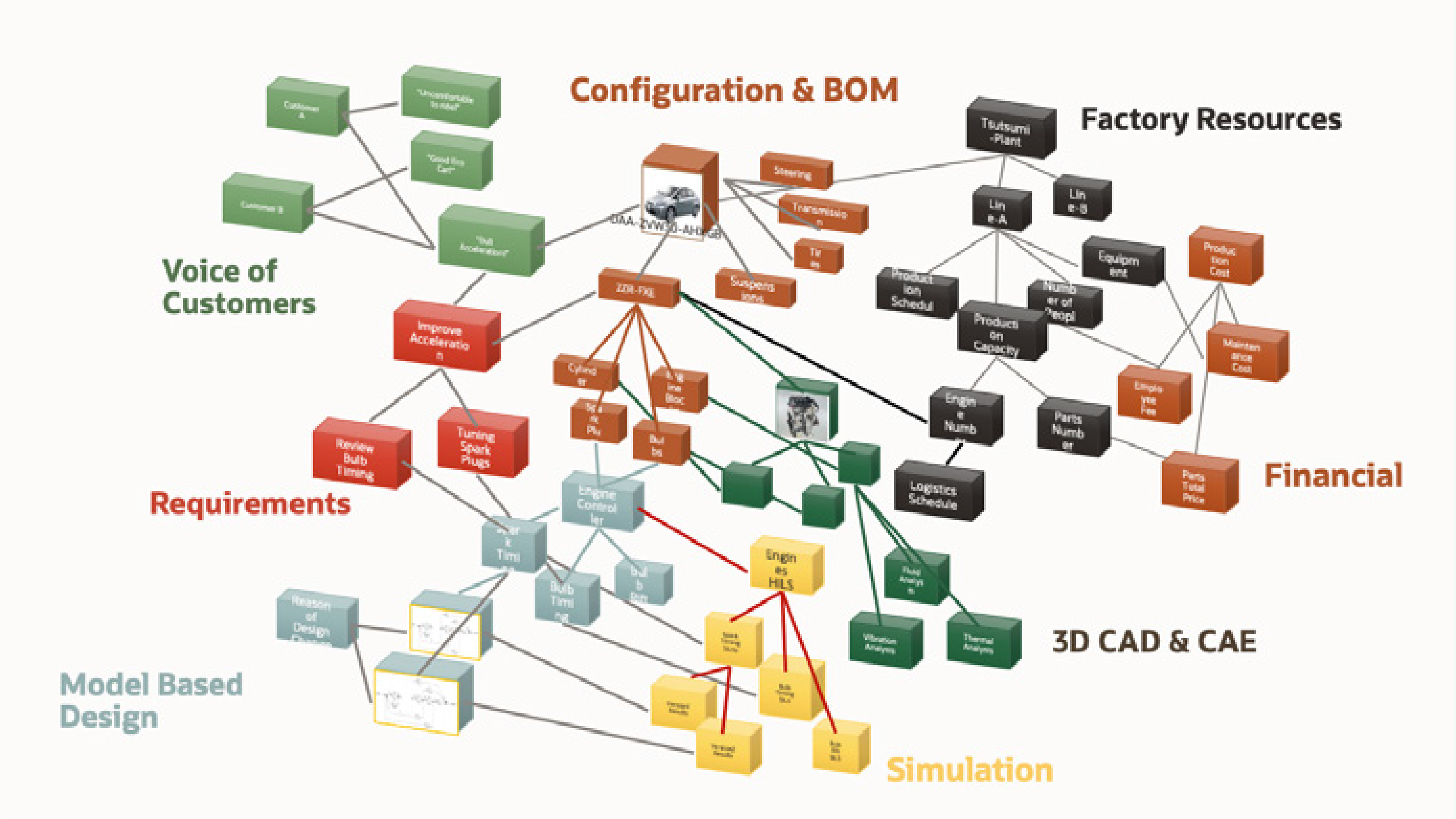
A car has 30,000 parts. So what is the impact of changing a part? What if you change a few parts at once? This kind of analysis can be very complicated with a car, where each part can have potentially thousands of dependencies. The queries for such analyses once took a significant amount of time because of all the needed multi-step table joins.

The graph solution

By using graph queries, the response time can be shortened to seconds or even faster, which means that real-time interactive analysis is now realistic. Graphs take the relationships that all the parts have with each other, and makes them clear—so that any flaws or negative dependencies also become clear.



By using a graph for a bill of materials analysis, you can create a model for analyzing the product information and dependencies. You can also add further information about the products, such as vendors, engineers, suppliers, materials, age of materials, etc. to create a variety of models. This enables you to explore component trust, vendor reliability, supplier options, and more.



Traceability

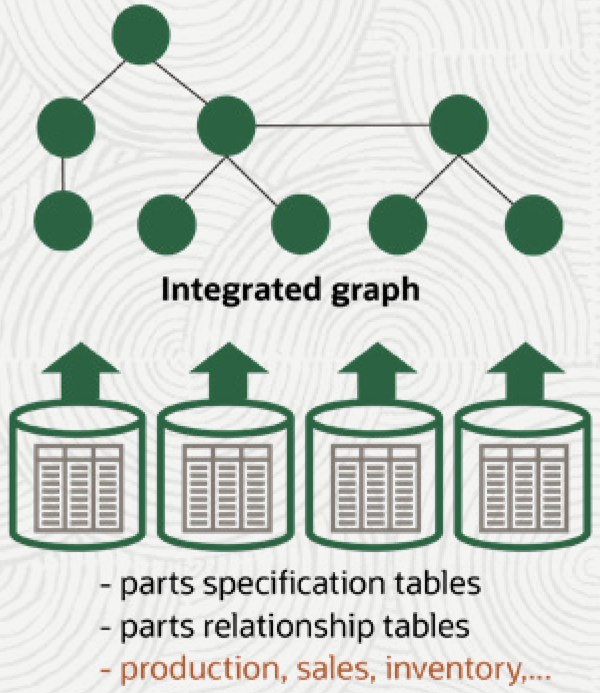
The problem

Traceability is of great significance in the manufacturing world. An automobile company might have to issue a recall for a car model because that specific model has a component which was produced from a factory during a limited time slot. The company must trace the causal components and then find the cars that are in market or on the road from the factory. This can be very difficult.

Most companies have a production database that manages the lot information on the product. But they also have a separate retail database, and a separate sales database, and a separate shipping database. It is complicated to discover all of the relevant information to find the cars with the problem, where they were shipped, and to whom they were sold.

The graph solution

Without graph technologies, analysts must combine all of these databases and run a traversal query from one specific car to the factory database that is managing the production line. All of this requires complex data modeling and many joins—unless the company has a graph database to connect all the relationships and graph algorithms to highlight connections and relevant information.


Integrated graph
- parts specification tables
- parts relationship tables
- production, sales, inventory,...

Master data management

The problem

At many manufacturing factories, the design team may use a certain name for a component. The production department might use another name. And other departments may use other names as well, all for the same item. When problems arise or when the company wishes to find out more about certain use cases and which components are involved for that specific item, all of the different, inconsistent names make it difficult to match to the right item and discover which components are the ones in question.

The graph solution

RDF graphs are good for modeling different components and utilizing the relationships and connections they have with each other. RDF graphs use all of this information to create a metadata layer that helps determine whether different names indicate the same item, whether the items are related, and even indicate whether different items can be used interchangeably because of their similarities. This is used in the pharmaceutical world as well, for identifying different chemicals, medicines, and generic names.

Without an RDF graph, applications usually embed logic to help find the correct items. But this logic doesn't always work across all databases, as each one often has different naming conventions. And if the DBA who created that application logic leaves the company, then the logic often gets lost.

Using an RDF graph to abstract the information into a metadata layer not only removes this reliance on application logic, but it also creates additional useful layers. RDF graphs don't just tell you whether different names refer to the same manufactured item; they also expose the relationships and dependencies these items have with other items, making it easier to find the other related items and discover implicit facts and relationships.

In a nutshell, with an RDF graph you gain a way to have self-describing data that captures content and semantics in a machine-readable and usable way. In addition, there is no need to ensure that the application logic is kept up to date; applications automatically get better as the RDF content and quality get better.

Government

From criminal activity to contact tracing, many government-related issues can be addressed with graph technologies.

Tax fraud

The problem

Tax fraud is a growing problem for many governments. The governments often become more resource-strapped while criminals grow more inventive. Not only that, but modern technology presents new challenges for less-agile governments and provides easy ways to move money across international borders, thus incentivizing criminals even further.

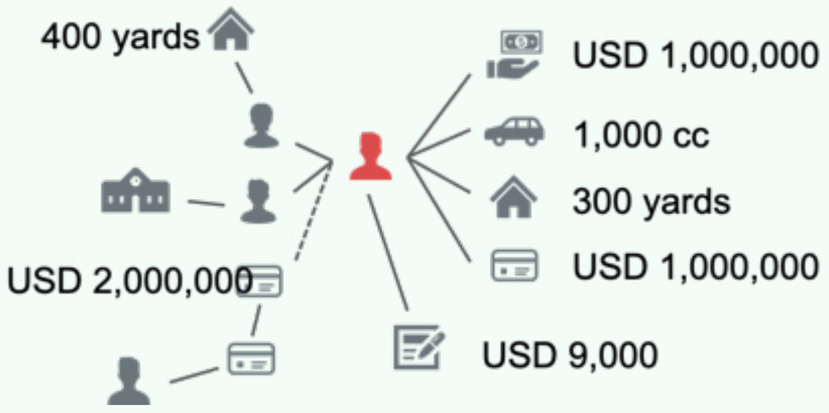
Now, criminals can set up shell corporations, then make these corporations look like legitimate entities. Money gets routed through multiple accounts, back and forth and all around, in a circuitous and deliberately confusing path that ultimately ends up with government money in the hands of criminals.

The graph solution

Untangling these complex paths is no easy task, with multiple layers of relationships hidden deep within data. Tracking the path through each layer of the relationship is a difficult task, but graph databases can help understand the structure of the shell corporate entities, provide visualization tools to help with manual investigation, discover suspicious patterns in multiple hops, and discover the paths that meander and ultimately lead back to one corrupt person or organization.

In a different tax-fraud use case, graph technologies can also uncover hidden properties and wages that people are trying to hide. For example, an individual may receive wages from several businesses and try to hide some of them. Or he or she may have other investment assets that weren't disclosed. And when there is income from multiple sources, including rental properties, royalties, partnerships, estates, and trusts, it can be difficult to track all of it and ensure that the right taxes are being paid, especially when there are multiple people involved in the ownership of the assets.

Graph technologies can lay out these assets and the people involved to make the relationships between them—and the money owed—clearer.



Criminal investigation

The problem

Graph databases are revolutionizing criminal activity analysis. Some crimes happen on a small, opportunistic scale. But the kind of crime that police work together to track and take down tends to happen on a large scale with many interconnected people, gangs, businesses, and even locations—which means that it doesn't tend to happen in silos.

The graph solution

Putting data into graphs provides a natural and efficient way of identifying criminal networks and looking for patterns. By applying graph-based algorithms like PageRank or centrality, it becomes easier to look for vulnerable people in the graph, discover more insights regarding locations, and even look for important people and potential criminal gangs. For example, by applying betweenness centrality, users can find the "weakest link," meaning the vertex that the graph relies upon. If you remove that vertex, the entire graph may fall apart—meaning you may have just found the linchpin of a criminal gang.

Contact tracing

The problem

Disease contact tracing is a critical activity worldwide. People become ill with highly infectious, new diseases and continue to live their normal lives, visiting movie theaters, packed gyms, crowded weddings, and choir practices—spreading that disease everywhere they go.

When someone is diagnosed, the race to find everyone who has been in contact with that sick person to ask them to quarantine, becomes a race against time. Contact tracers must be able to do their work as quickly as possible to stop further spread of the disease.

The graph solution

Graph databases, with their heavy emphasis on relationships, are ideal to use for analyzing disease patterns. Analysts can input information on the people who have tested ill, the family members and friends they've interacted with, and the places they've visited, to rapidly locate hotspots and connections. In this way, analysts can work more quickly to isolate those who are ill and prevent further disease outbreaks.

There are three levels to contact tracing with graphs.

First, there is the need to understand people's relationships, communities, and the places they visit which graphs can make clear if provided with enough mobile data.

Secondly, graphs must find the possible spread—which means looking at potential links between people who might spread the disease. Did the person travel by bus? Can we identify everyone on the bus?

Third, contact tracers must find "super spreaders" and rush to isolate those people first. This involves finding the people who have wide and dense contacts and who are likely to have links to many different communities. This involves exploring the graphs with notions of centrality and betweenness, to find the highly connected people.

Data regulation and privacy

As data becomes more valuable, companies more actively collect, sell, and utilize it. At the same time, the laws, regulations, and standards around data have increased significantly as well. But as data continues to increase in volume, managing that data and ensuring data privacy and regulations become ever more complex.

GDPR

The problem

Data management professionals everywhere are still grappling with the problem of addressing GDPR. How can they continue to maintain people's privacy, respond to data access requests, and fulfill requests to be forgotten, among other issues?

One of the major difficulties lies with discovering what is stored in each database. Data is moved around. Data is transformed. Data can be consumed by users and other processes. And it can be extremely difficult to track and trace what has happened with all of this data.

But that's not the only problem. The data may have originally been stored in a table—but then reports were created from the data. The reports contain information and have access rules as well. If someone wants to exercise their right to be forgotten, tracking that electronic trail of where the data originally lived, where it was copied to, and where it was used in tables and reports—all of that becomes extremely complicated. Fulfilling the requirements of that element of GDPR is a monumental task

The graph solution

Tracking data lineage is a perfect match for a graph. The various steps in the data lifecycle can be tracked and navigated, vertex by vertex, by following the edges. With graph, it becomes possible to follow a path and see where the information originally resided, where it was copied, and where it was utilized. With all of this information laid out in a graph, it becomes simpler for data professionals to determine how to fulfill GDPR requests and remain compliant.

Data privacy

The problem

Organizations need to limit access to data. For example, perhaps they only want to allow certain personal computers to open certain files. Or they want certain teams, departments, and projects to have access to certain data. Access rights are complicated to manage and visibility into which teams have access, which teams shouldn't have access anymore, and which teams need better access to perform their jobs, can be very complicated.

Often, this data structure needs to be fluid so the hierarchical structure can change dynamically. But doing this seamlessly is difficult, and true insight into what is being changed and how it is changed is hard to achieve.

The graph solution

Graph can make such a hierarchical structure very dynamic and graph query can enhance the response time for changing data access.

Because of the complex and dynamic access controls, the applications have to check the permission of a specific material every time. But graph queries can follow the network so efficiently that the applications can find the permissions in real-time.

Cyber security

The problem

Cyber security is a very important topic in the battle for cloud. This involves complicated areas such as invalid traffic detection, cyber threat hunting, and malware detection. One solution for addressing these topics is using graph technology to enhance cyber security.

The graph solution

Graph technologies capture connections between data entities, i.e., how computers are connected over an IT network. They exploits extra signals from graph for anomaly detection. They can enhance cyber-threat detection by enabling interactive, visual exploration of security data. This creates an ideal environment for cyber-threat hunting.

In one example, Oracle collaborated with a SaaS security team to enhance a threat intelligence system. The system would monitor application executions, detect suspicious activities, generate incident alerts, and assign the issue to be investigated. Often, these issues are manually investigated which takes time and can be an inefficient use of resources.

Certain incidents would always follow a pattern, however. By adding graph-based visual threat investigation to track the patterns of where information packets were coming from and how they were being forwarded, these consistent patterns could be automatically identified and stopped, thus saving time and employee resources.

Marketing

Marketing is all about relationships and marketers must understand their customers, the relationships their customers hold with each other, products, relationships between different products, and much more to effectively provide customers with what they want.

Customer 360-degree analysis

The problem

Today, companies have increasingly more information about customers, including:

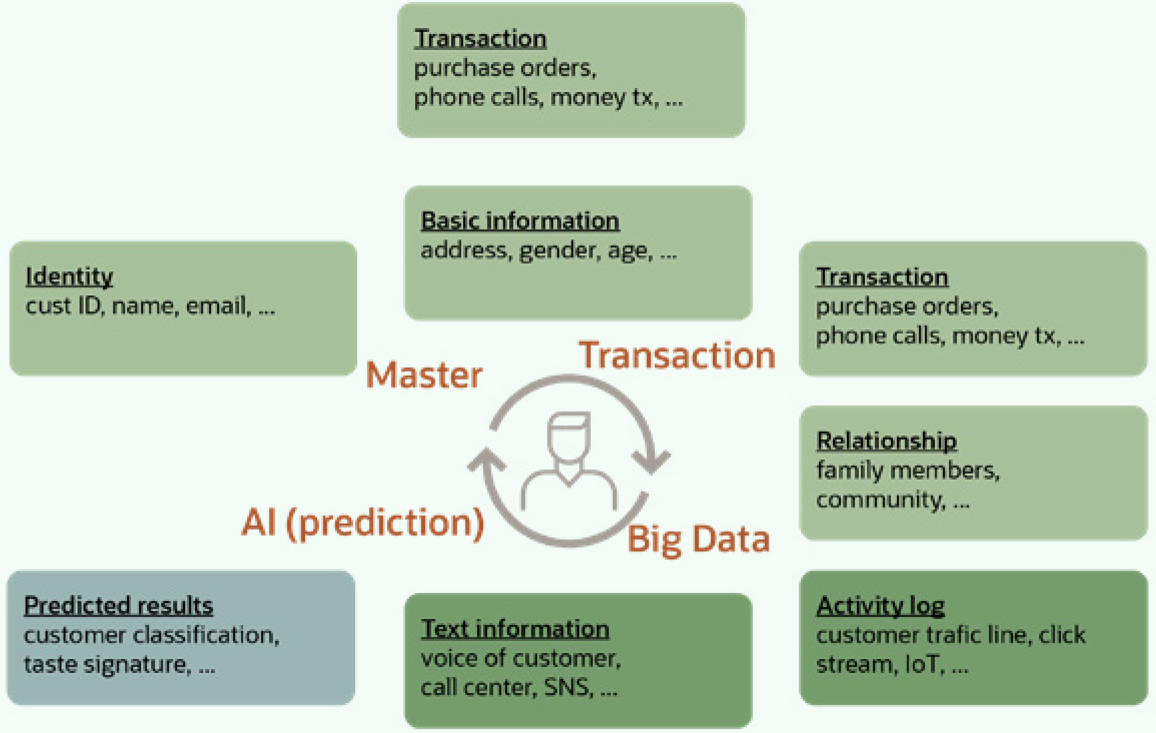
- Master data — name, age, gender, address
- Transactions — purchases, types of items bought, purchase times
- Big data — call center logs, traffic lines, web click streams, SNS activities
- Predictions — classification, taste signatures (often created by different models)

But the companies often don't use this information as comprehensively as they should. Creating a true customer 360-analysis is difficult.

The graph solution

When all of the marketing data listed above is collected and integrated into the physical platform, it is usually difficult to analyze all of it. But these datasets can be logically integrated on graphs and the graph users can simply view all of the surrounding information of one entity (the customer). With graphs, marketers can gain a more comprehensive view of their customers—the relationships the customers hold with each other, the relationships between all the purchased products, and more. Then, graph users can run algorithms to discover even more fine-grained detail about the customer.

Viewing all information about one specific customer is important to understand the customer and to perform customer 360-analysis, to discover which predictions (usually created through machine learning) are true and why.



Transaction
purchase orders, phone calls, money tx, ...
Identity
cust ID, name, email, ...
Basic information
address, gender, age, ...
Transaction
purchase orders, phone calls, money tx, ...
Master
Transaction
Relationship
family members, community, ...
AI (prediction)
Big Data
Predicted results
customer classification, taste signature, ...
Text information
voice of customer, call center, SNS, ...
Activity log
customer trafic line, click stream, IoT, ...

Product recommendations

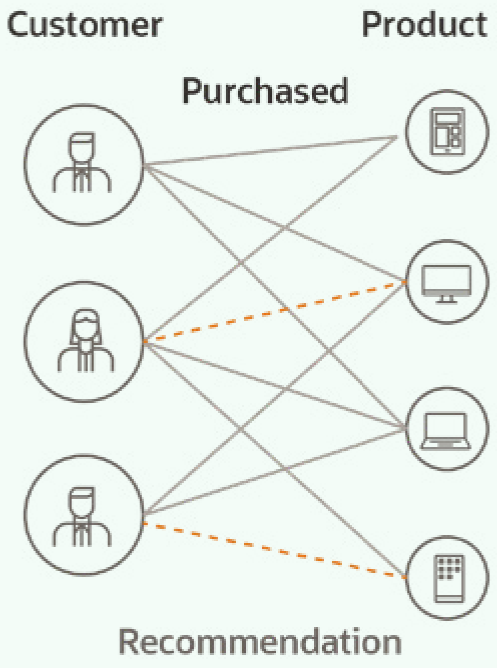
The problem

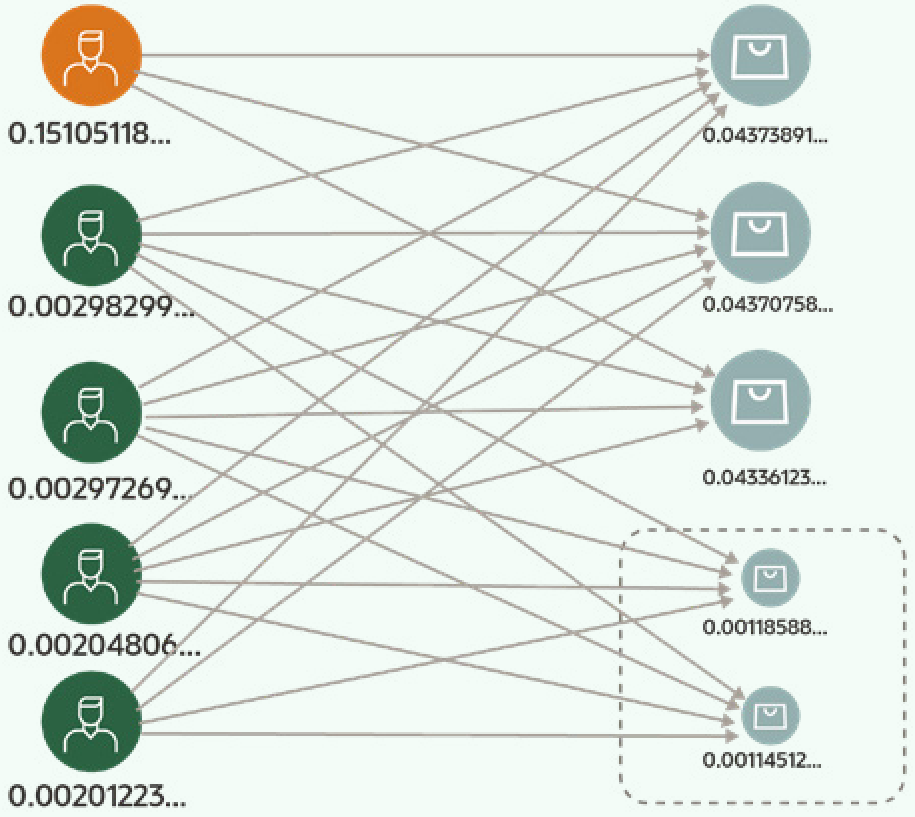
Non-graph technologies can support recommendation engines, but graph creates faster time-to-value. Graph databases are built so the relationships between customers and the products they like to buy are already laid out—so it becomes easy and fast to run algorithms through the data to find recommendations.

In addition, real-time recommendations are becoming more important than ever. But this requires the ability to correlate product information, customer inventory, past customer behavior, current supplier information, logistics, and even social data such as ads clicked and products explored via social media. This is extremely difficult for certain kinds of databases.

The graph solution

The technology for collecting all of this data and forming connections to gain speedy insight into customer needs and product trends—and then to provide real-time recommendations—is the graph database. In fact, many large corporations rely upon graph analytics to provide their recommendations because the relationships are already laid out, and analysis of these relationships to provide recommendations is very fast.





Social media

The problem

Social media is a rising part of our world today and relationships are a key part of it. So is connecting users—and ensuring the validity of those users. In the social media world, “sockpuppet” accounts are an issue. Sockpuppets are fake accounts run by bots. They work to make certain topics or keywords look more important by liking or sharing them, and thus making them look like they’re trending.

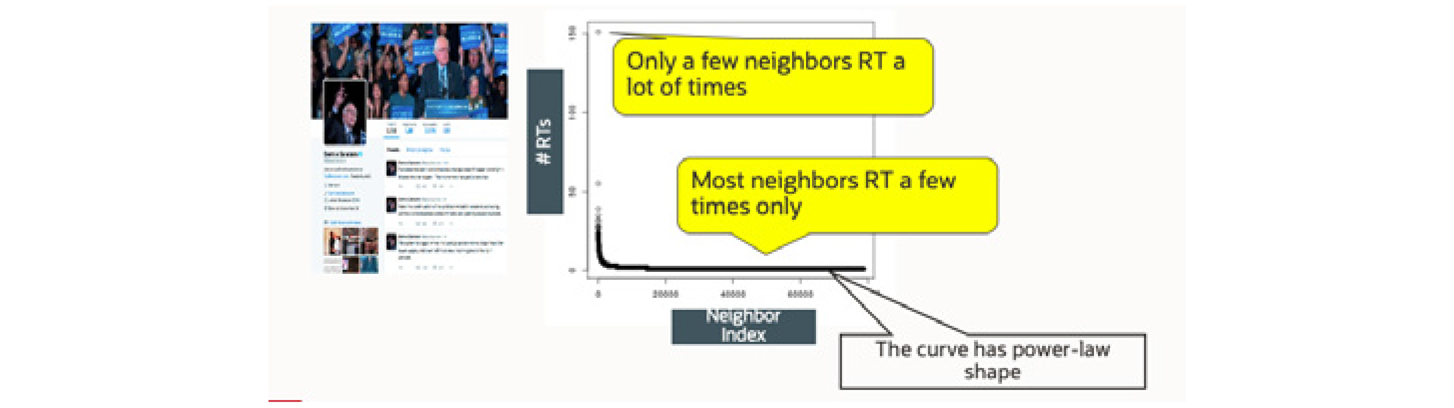
At times this is fairly innocuous, although still deceptive to retailers and customers. Think of an Instagram influencer purchasing followers and likes to make himself or herself look more popular. At other times, this can be very serious, with countries using bots to encourage trending topics that destabilize other governments.

The graph solution

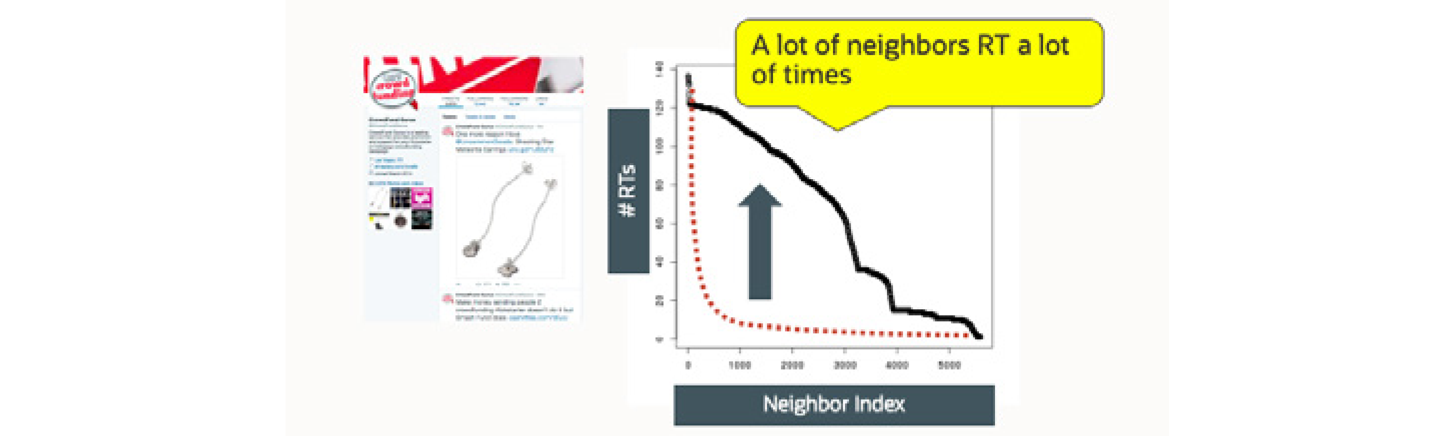
Graph databases can traverse social networks and related data very quickly, which is why social media companies such as Facebook, LinkedIn, and Twitter all leverage some kind of graph processing within their platform to identify friends and families across the world. We already mentioned product recommendations in the previous example. A similar process can be used to provide recommendations of users, images, products, and more—and also for detecting fraudulent activity and sockpuppet accounts.

In this example, we created a graph between accounts with retweet counts as edge weights to see how many times these accounts are retweeting neighboring accounts. The displayed pattern tends to exhibit very different characteristics from naturally popular accounts.

Here is the pattern for a naturally popular account:



And here is the pattern for an unnaturally popular account:



With graph, it’s possible to quickly spot unnatural behavior and weed out the bots and sockpuppet accounts.

AI and machine learning research

AI and machine learning are commonly seen as areas of great interest because of their promise in improving business results and creating new impact. Graph can be used to augment data science in a few key ways.

Feature engineering

The problem

When it comes to machine learning, machine learning models rely upon data. The better this data—the richer, deeper, and more complete it is—the better the machine learning model (usually). There is an entire step to creating a machine learning model, called feature engineering which involves enriching the data. Here's a simplified example: a data scientist might have a person's home address and office address—but having the distance in miles would be better for the machine learning model. The data scientist must perform this extra step of feature engineering to find that distance in miles and create data that's better for his or her machine learning model.

However, there are certain kinds of feature engineering that can be more complicated to accomplish—especially when it comes to looking at relationships to data and bringing those relationships to the forefront. Trying to do so can require too many joins and be slow and cumbersome to accomplish.

The graph solution

Most commonly, features for machine learning can be created via graph by running graph algorithms on a dataset that has been loaded into a graph database, and creating enriched data which can then be used for machine learning. This step of feature engineering provides the machine learning model with more comprehensive, useful information.

For example, a machine learning model might already have information about a new customer who is looking to buy life insurance, including where that person lives and which car is owned. The model may classify existing customers and based upon that, make predictions for the new customer. But it would be missing the graph component. Perhaps the new customer has coworkers who are already customers and that can be a key indicator for why the customer might actually sign up for life insurance. By including features derived from graph, machine learning models can become more powerful and more accurate.

Alternatively, graph algorithms can be run on data to generate new insights, such as using clustering to find similar customers based on the products they bought.

Graph neural networks

The problem

We've already discussed how graphs can help with recommendations—but what about predictive recommendations? For example, what if an online retail store wants to send recommendations to a customer, with timing determined by when the customer is predicted to run out of the item? Adding predictions to recommendations can be complicated but it can also greatly increase profits. It is often an area of untapped opportunity for many businesses.

The graph solution

Many data scientists are starting to become interested in graph neural networks, which can capture the graph itself as an input of machine learning and neural networks. The graph can potentially hold more information than standard tables because of the flexibility of the model. Machine learning models with information captured from graphs often provide better performance than machine learning based on table shape input.

This type of neural network is already being evaluated across industries, and some results show that it improves accuracy e.g. in financial fraud detection. For running such techniques, keeping the original information in graph format for more flexibility is essential, so the graph database is the key component to build workflows that utilize state-of-the-art machine learning techniques.

Why graph technology from Oracle?

In today's hyper-competitive world, new ways to use the data you already have are constantly emerging.

Graph, a technology you may not have been aware of, has been changing and enhancing analytics. It is a powerful tool you can use to solve problems and obtain valuable insights faster. By seeing what graph technology can do for you, you can make better use of your data, find results faster, and help ensure your organization stays current.

Oracle makes it easy to adopt graph technologies in your enterprise. A graph analytics engine is included with Oracle Database and Oracle Autonomous Database so users can discover more insights in their data by using the power of graph algorithms, pattern matching queries, and visualization.

All of the use cases listed in this ebook are real-world examples and have already been implemented on Oracle across several industries. Many can only be accomplished with Oracle's graph technologies and their unique and enterprise-grade capabilities.

Although all graph databases claim they are high-performance, only Oracle's graph offerings are performant both in query performance and algorithms, as well as tightly integrated with an industry-leading database. This makes it easy for developers to add graph analytics to existing applications and make sure of the scalability, consistency, recovery, access control, and security that the database provides by default.

If you're interested in finding out more, get started today with our step-by-step guide and try graph technology in Autonomous Database with Oracle's Always-Free Tier.

Learn more

Visit the webpage

Try graph for free

ORACLE