GRAPH+AI SUMMIT

organized by TigerGraph

Building a Comprehensive Data Lineage Solution with a Graph Database

Robb Horton - Sr. Sales Engineer

April 2021

Today’s Presenter

Robb Horton

Senior Sales Engineer, Tigergraph

- BS in Systems Analysis from Miami University
- 20+ years in Business Intelligence and Data Warehousing
- Located near Cincinnati, OH
- 7 children and 2 grandchildren (so far)

What is Data Lineage?

Data lineage includes the data origin, what happens to it and where it moves over time.[1] Data lineage gives visibility while greatly simplifying the ability to trace errors back to the root cause in a data analytics process.[2]

- Represented visually
- data flow/movement from its source to destination
- data gets transformed along the way,
- how the representation and parameters change,
- how the data splits or converges after each hop.

WIKIPEDIA
The Free Encyclopedia

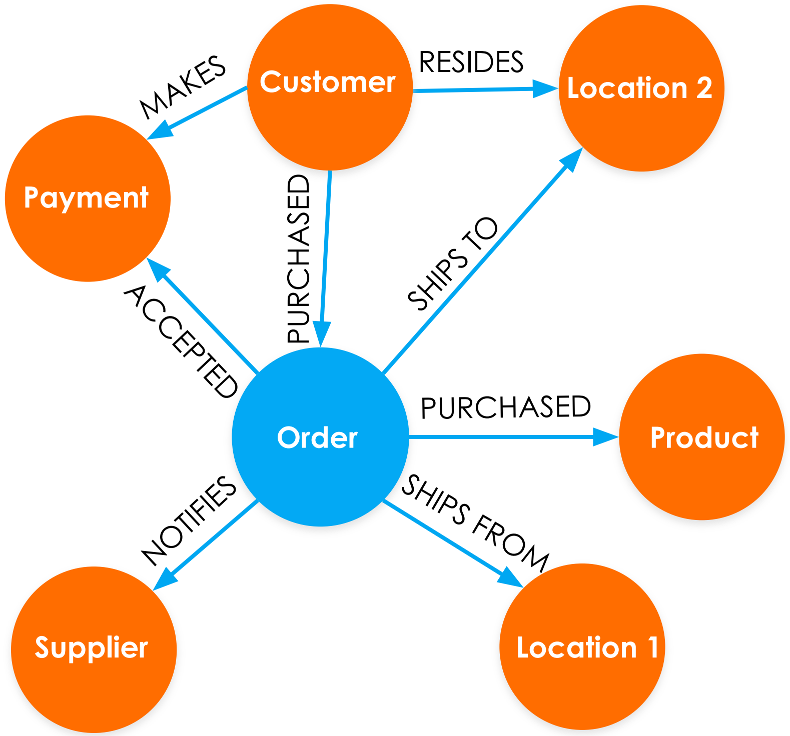
“A simple representation of the Data Lineage can be shown with dots and lines, where dot represents a data container for data points and lines connecting them represents the transformations the data point undergoes, between the data containers.”

- aka “a graph database”

Why Do We Need Data Lineage?

- Operational Intelligence
- Consistency of Business Terms
- Root Cause Analysis/Remediation
- Impact Analysis
- Performance Assessment
- Policy compliance
- Auditability

Why Graph Analytics, Why Not RDBMS?



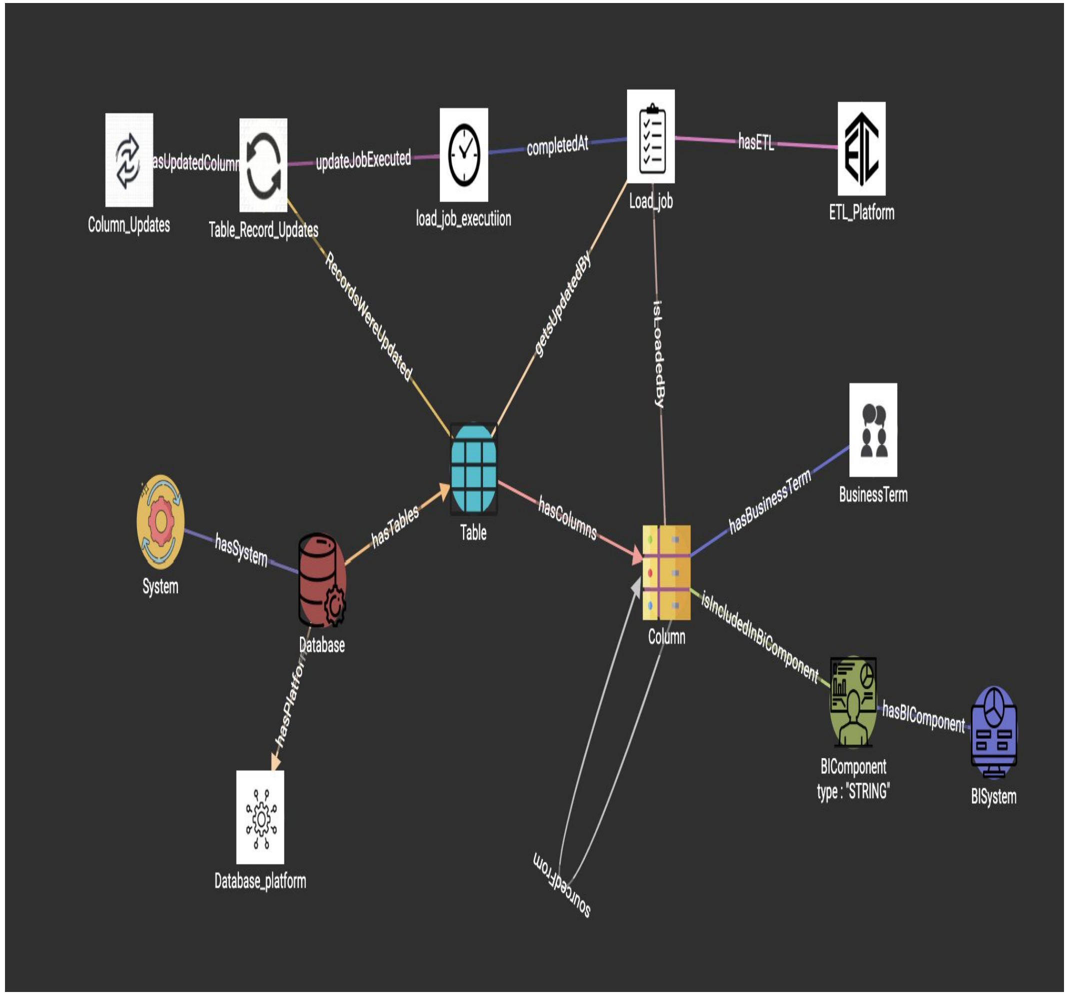
- **Definition** - Graph analytics is a set of analytic techniques that allows for the **exploration of relationships** between entities of interest such as organizations, people and transactions.
- **Forecasted growth** - 100% annually through 2022
- **What's driving the growth**
    - Need to ask **complex questions across complex data**, which is **not always practical or even possible at scale using SQL queries.** (RDBMS requires time-consuming & expensive table joins!)
- **What's needed for broad adoption of graph data stores**
    - Graph data stores can efficiently model, explore and query data with complex interrelationships across data silos, but the **need for specialized skills has limited their adoption to date.**

Graph deployments are going deeper, wider and operational:
Need to make it accessible to non-technical users

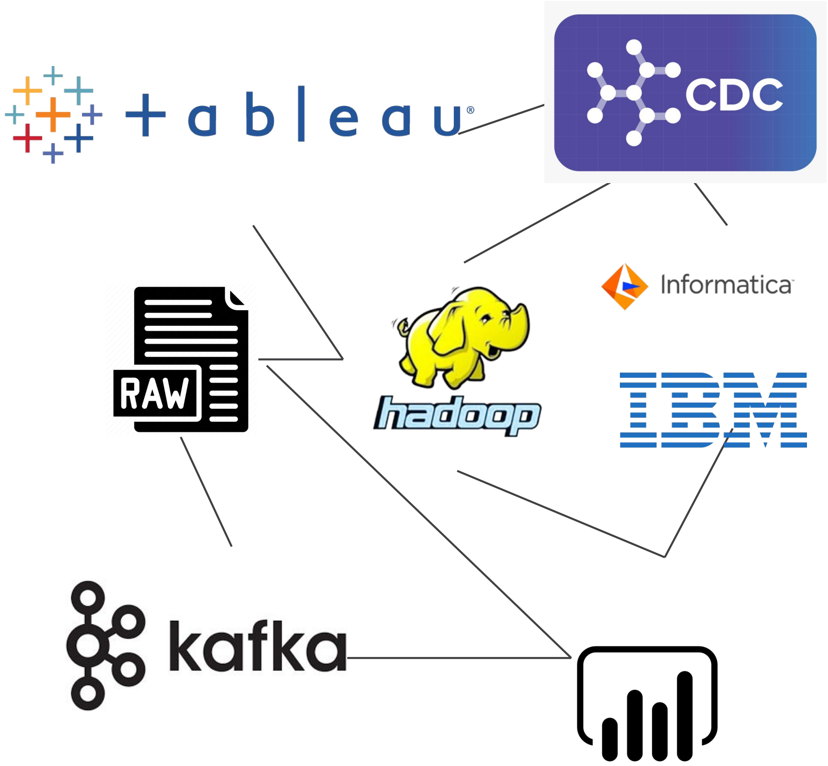
Source: Gartner - Top 10 data and analytics Trends for 2019

Why Graph for Data Lineage?

- Lineage Data is highly connected data
- Lineage depth is unknown, and can vary for each entity
- GSQL queries simpler to write and read than relational SQL for variable depth queries
- Graph allows for the data easily change, with new types of nodes and edges



Where do we get “Data Lineage”?



Data Connectivity

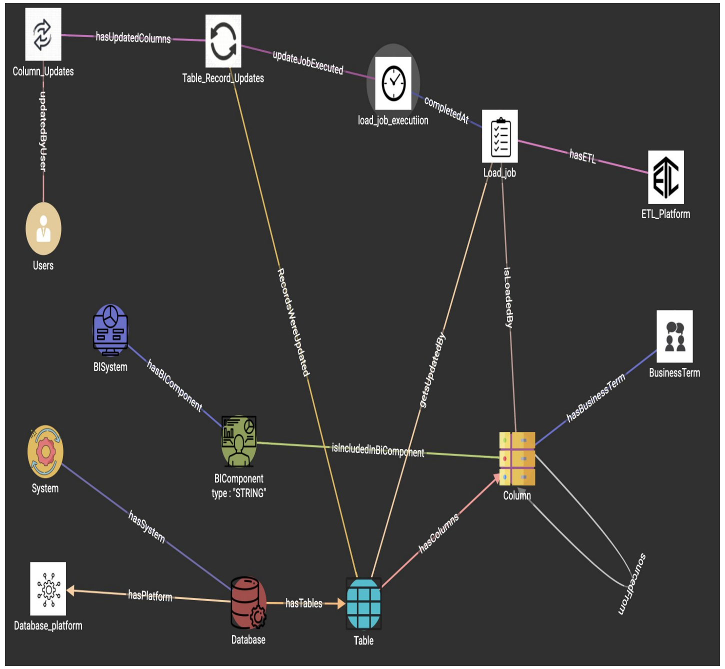
(1) Diverse Data Sources
(2) Not Interlinked
(3) Privacy & Security on Connectivity
(4) Hence,No Ad-Hoc Querying
(5) Poor Performance on Speed And Scale

Data Quality

(1) Diverse Data Formats
(2) Millions of Entities
(3) Thousands of Attributes
(4) Data Consistency

Connected Data
Perform at Scale
Perform at Speed
Cover Future Needs

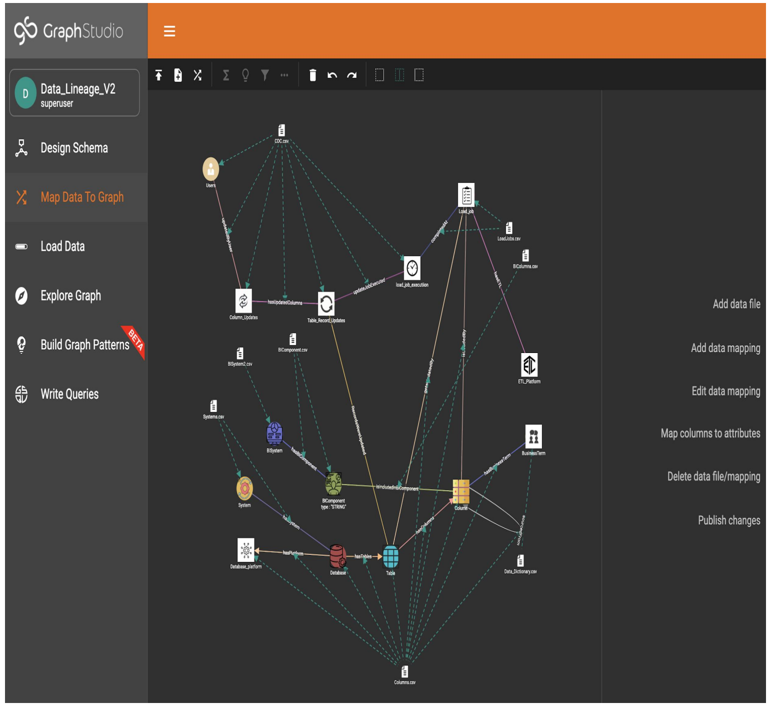
Choosing Graph

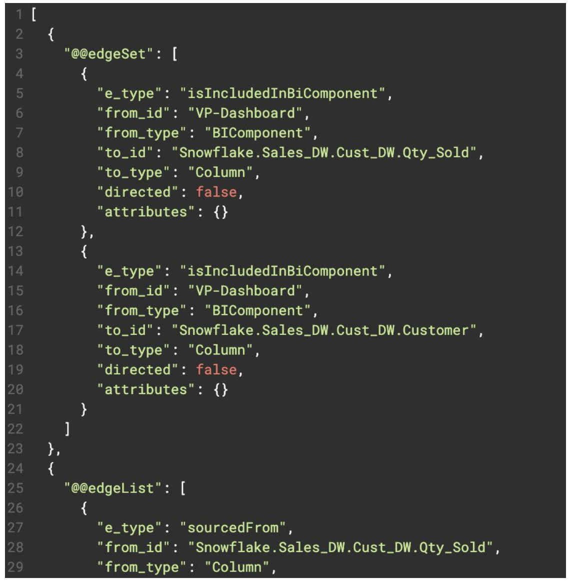


Download the solution brief at - https://info.tigergraph.com/MachineLearning

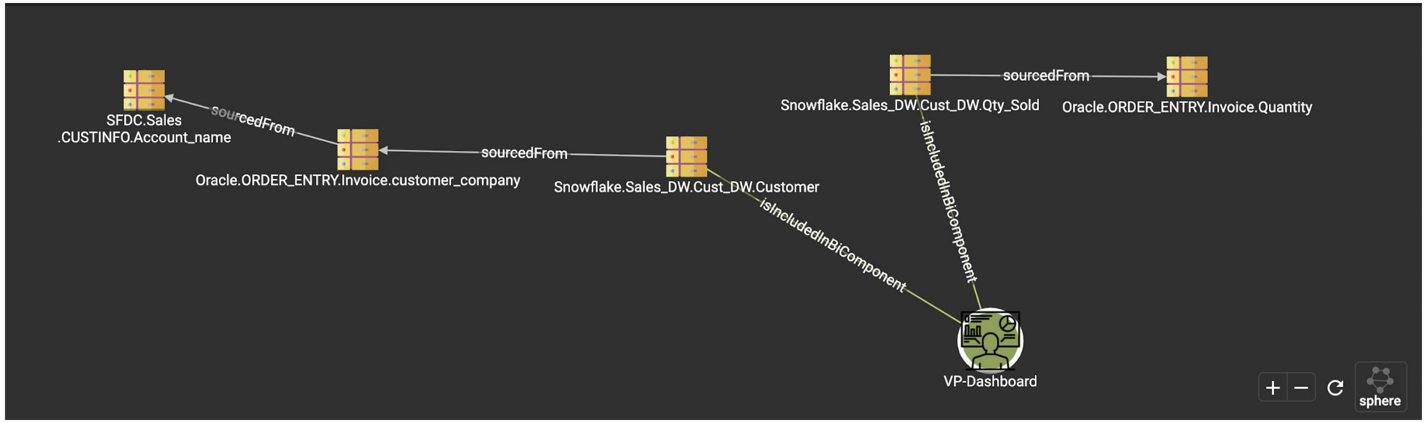
Our Demo

- GraphStudio
- Small dataset
- AWS RHEL instance 8GB/40GB





```
[
  {
    "@@edgeSet": [
      {
        "e_type": "isIncludedInBiComponent",
        "from_id": "VP-Dashboard",
        "from_type": "BIComponent",
        "to_id": "Snowflake.Sales_DW.Cust_DW.Qty_Sold",
        "to_type": "Column",
        "directed": false,
        "attributes": {}
      },
      {
        "e_type": "isIncludedInBiComponent",
        "from_id": "VP-Dashboard",
        "from_type": "BIComponent",
        "to_id": "Snowflake.Sales_DW.Cust_DW.Customer",
        "to_type": "Column",
        "directed": false,
        "attributes": {}
      }
    ]
  },
  {
    "@@edgeList": [
      {
        "e_type": "sourcedFrom",
        "from_id": "Snowflake.Sales_DW.Cust_DW.Qty_Sold",
        "from_type": "Column",
```


SFDC.Sales
.CUSTINFO.Account_name
sourcedFrom
Oracle.ORDER_ENTRY.Invoice.customer_company
sourcedFrom
Snowflake.Sales_DW.Cust_DW.Customer
Snowflake.Sales_DW.Cust_DW.Qty_Sold
sourcedFrom
Oracle.ORDER_ENTRY.Invoice.Quantity
isIncludedInBiComponent
isIncludedInBiComponent
VP-Dashboard
sphere

TigerGraph

Get Started for Free

- Try TigerGraph Cloud
- Download TigerGraph's Free Enterprise Edition
- Take a Test Drive - Online Demo
- Get TigerGraph Certified
- Join the Community

@TigerGraphDB /tigergraph /TigerGraphDB /company/TigerGraph