

Graph Analytics for SQL Developers - The New MATCH Clause in SQL:2023

Dr. Abi Giles-Haigh – Field CTO, Oracle EMEA

Jim Czuprynski – Chief StoryTeller, Zero Defect Computing, Inc.



Futu Tech

Ire & P Casts:	ast Our FAV Features OAC Presented by Abi Haigh, Dan Vlami Van Sluys, Philip St Archive	19th of Ciles- is, Wayne pe Lions New Features i Oracle DB 23C Presented by Cathye Pendley and Gerald Venzl Click	n Unveiling the Mystery: A Beginner's Guide to Machine Learning Interpretability (MLI) Presented by Sai Nikhilesh Kasturi	c F P	Dec 14th Dur FAV eature of OAC esented by TBD
2023	2022	2021	2020		2019
Date Title		Presenter	(s)	Replay	Download(s)
May 4 Oracle	APEX: A Swiss Army Knife Story for N	our Analytics Lucas Hir	schegger & Simon Collins	Video	Slides
Apr 20 From [oata to Insights with Oracle Analytics	Joel Acha		Video	Slides
Apr 6 Data P	atform Migrations – Few Learnings	Sujata Ba	upala & Sanjay Sabnis	Video	Slides
Mar 14-16 AnDOL	C Summit 2023	AnDOUC			
Feb 16 Favorit	e New Features in Jan 2023 OAC				
Jan 26 Summ	t Preview of Presentations	Various P	esenters	Video	Slides

Nov 2nd

Oct 19th

Nov 30th

Submit a topic to share at https://andouc.org/techcasts/









We Have Merch!

Show your "Tech Side" in everything you do! Visit the AnDOUC Store at ANDOUC.ORG











Let's Connect



Website http://andouc.org/



Chat with the Experts https://bit.ly/Join-ANDOUC-Slack



Watch Previous TechCasts https://bit.ly/3qmGgHN



@AnalyticAndData

https://www.facebook.com/ AnDOracleUserCommunity https://www.linkedin.com/c ompany/analytics-and-dataoracle-user-community



Spatial + Graph SIG bit.ly/Spatial-Graph-LinkedIn

Introduction

- PhD in ML Computation Modelling of the human heart
- Oracle DBA for 5 years
- Worked public health care, private banking
- Oracle ACE Director Previously.....





Who Am I, and What Am I Doing Here?





Modern Apps Need To Generate Value From Data in New Ways





Why would you use Graph Databases?

Graphs are useful when **discovering** and understanding relationships is important





Why would you use Graph Databases?

Graphs are useful when **discovering** and understanding relationships is important

For example, with this data...

Customers		Accounts		Transactions				
cst_id	name	acc_id	cst_id		src_acc	dst_acc	amount	date
c1	Alice	al	c2		al	a3	\$20,000	2020-10-01
c2	Bob	a2	c1		a5	a4	\$30,000	2020-10-02
c3	Charlie	a3	c4		a4	a2	\$10,000	2020-10-03
c4	Dave	a4	c3		a3	a5	\$20,000	2020-10-04
		a5	c1					



Why would you use Graph Databases?

Is there **any money flow** between Bob & Charlie?



Graph-based queries:

- Fast traversals
- Path finding
- Identify patterns
- Extract subgraphs





How to identify the clusters of transactions?



Graph algorithms:

- Community Detection
- Ranking and Centrality
- Paths and Connectivity
- Link Prediction, Similarity

Use case: BOM (Bill-of-material) Analysis in Manufacturing

Manage complex hierarchies of parts and components

Analyze nested relationships using very compact queries, e.g.

- Aggregate required parts for a product variant
- Analyze dependency on a given part

Similar use cases:

- Data lineage
- GDPR compliance





Use case: Fraud Prevention in Financial Services

Manage large networks of interconnected entities

Detect suspicious patterns in financial transactions, eg.

- Indirect connections to suspicious/fraudulent accounts
- Circular money flows, indicative of money laundering
- Layering (hiding money flows in large number of small, indirect transactions)





Analyzing relationships, direct or indirect







Networks

Hierarchies

Paths

Patterns



Example – Financial Transactions





Bank Accounts and Cash Transfers



BANK_ACCOUNTS

Property Graph Data Model: Key Elements

A set of vertices (also referred to as nodes) representing entities

A set of **edges** that connect vertices representing relationships







Property Graph Data Model: Key Elements

A set of vertices (also referred to as nodes) representing entities

A set of **edges** that connect vertices representing relationships

Vertices and edges can have

• Properties (zero or more), a typed key/value pair

A graph can have different types of vertices and edges







BANK_ACCOUNTS



FROM_ACCT_ID	TO_ACCT_ID	DESCRIPTON	AMOUNT
1	672		1000
1	584		1000
1	259		100000
2	833		5001
2	840		7050
2	493		4363



BANK_ACCOUNTS



FROM_ACCT_ID	TO_ACCT_ID	DESCRIPTON	AMOUNT
1	672		1000
1	584		1000
1	259		100000
2	833		5001
2	840		7050
2	493		4363



BANK_ACCOUNTS





BANK_ACCOUNTS

BANK_TXNS



FROM_ACCT_ID	TO_ACCT_ID	DESCRIPTON	AMOUNT
 1	672		1000
1	584		1000
1	259		100000
 2	833		5001
2	840		7050
2	493		4363

BANK_ACCOUNTS table is a vertex table

• Every row is a vertex

BANK_TXNS table can be an edge table

- Every row is a cash transfer
- Every cash transfer is an edge between two vertices (accounts)









Are there highly connected accounts?





Are there highly connected accounts?

Are there cycles starting with acct_id 259?





Are there highly connected accounts?

Are there cycles starting with acct_id 259?

Are there paths between acct_id 681 and 228?





Are there paths between acct_id 681 and 228? Are there least connected accounts?

Are there highly connected accounts?

Are there cycles starting with acct_id 259?



Creating a Graph from Relational Tables



Mapping relational tables to a graph

More than one mapping possible

- Model depends on planned queries and analyses
- Different from relational model
- N:m relationships are no issue Definition of graph in PGQL DDL
- VERTEX TABLES clause and EDGE TABLES clause list the source tables
- Defaults are taken from Oracle Data Dictionary

No data copy, data remains in place

- Like views on relational tables
- Particularly useful for operational workloads (DML plus query)

```
CREATE PROPERTY GRAPH bank_graph
    VERTEX TABLES (
        BANK ACCOUNTS as ACCOUNTS
        LABEL ACCOUNTS
        PROPERTIES (ACCT ID, NAME)
    EDGE TABLES (
        BANK TXNS
        SOURCE KEY (FROM_ACCT_ID)
            REFERENCES ACCOUNTS (ACCT ID)
        DESTINATION KEY (TO ACCT ID)
            REFERENCES ACCOUNTS (ACCT ID)
        LABEL TRANSFERS
        PROPERTIES (AMOUNT, DESCRIPTION) )
 OPTIONS (PG VIEW)
```

Specification (PGQL 1.5) https://pgql-lang.org/spec/1.5/#creating-a-property-graph



Querying property graphs



Graph Pattern Matching: MATCH Clause Syntax

Use () to represent vertices

- (a is person) variable a, label person
 Use [] to represent edges
- [e is owner] variable e, label owner Edge patterns
- -[]-> outgoing edge
- <-[]- incoming edge
- -[]- anydirected edge

Variable-length paths

- -[...]-> * zero or more hops
- -[...]-> + one or more hops
- -[...] -> {2,4} two to four hops
 Use ON to specify graph (optional)

SELECT a,b,e FROM MATCH (a)-[e]-(b) ON graph SELECT a,b,e FROM MATCH (a)-[e]->(b) ON graph

```
SELECT a,b,e FROM
MATCH (a is ACCOUNT)-[e is TRANSFER]->(b)
ON graph
```

SELECT a,b FROM
MATCH (a is ACCOUNT)-[is TRANSFER]->{1,3} (b)
ON graph WHERE a.account_no=4711

SELECT a,b,c,e1,e2,e3 FROM
MATCH (a)-[e1]->(b)-[e2]->(c)-[e3]->(a)
ON graph



Client tools

SQL Developer

• PGQL Worksheet

SQLcl

• PGQL plugin

Java CLI

Python CLI/Jupyter

Graph visualization tool





Support for SQL/PGQ in Oracle 23c



Native property graph support in SQL engine

- Metadata object over original data
- No data copy

MATCH clause inside GAPH_TABLE function

 Implementing SQL:2023 (ISO/IEC 9075-16)

CREATE PROPERTY GRAPH is the same

Combination with JSON possible

- JSON document as property
- JSON element as property



```
SELECT DISTINCT * FROM GRAPH_TABLE
(SH_PURCHASE
    MATCH (c1)-[is purchase]-> (p1) <-[is purchase]-(c2)
    WHERE c1.cust_last_name = 'Weasley'
    AND p1.prod_name = 'Pixel 13'
    COLUMNS (c1.cust_last_name as person_name,
        p1.prod_name,
        c2.cust_last_name as customer_name)
)
ORDER BY customer_name;</pre>
```

Using the MODEL Clause: A Demonstration



Using Graph Algorithms



Graph analytics: 60+ parallelized, in-memory algorithms out-of-the-box



Detecting communities

Strongly Connected Components, Weakly Connected Components, Label Propagation, Louvain, Conductance Minimization, Infomap



Ranking and walking

PageRank, Personalized PageRank, Degree Centrality, Closeness Centrality, Vertex Betweenness Centrality, Eigenvector Centrality, HITS, Minimum Spanning-Tree (Prim's), Breadth-First Search, Depth-First Search, Random Walk with Restart



Topology analysis

Conductance, Cycle Detection, Degree Distribution, Eccentricity, K-Core, LCC, Modularity, Reachability Topological Ordering, Triangle Counting, Bipartite Check, Partition conductance



Path-finding

Shortest Path (Bellman-Ford, Dijkstra, Bidirectional Dijkstra), Fattest Path, Compute Distance Index, Enumerate Simple Paths, Filtered and Unfiltered Fast Path Finding, Hop Distance



Link prediction and others

Twitter Whom-to-follow, SALSA, Adamic-Adar Index



Machine learning

DeepWalk, Supervised GraphWise, Unsupervised GraphWise, Pg2Vec, Matrix Factorization, GNNExplainer



Wrap-up



Summary

Graphs offer a different view on existing data

Oracle Graph is the graph database for the enterprise

Graph Studio in Autonomous Database makes it easy to get started





Additional Resources

Oracle Graph Resources

- Product Page:
 - www.oracle.com/database/graph
- Oracle LiveLabs
 - <u>bit.ly/GraphLiveLabs</u>
- Blog: Examples, tips and tricks
 - <u>bit.ly/OracleGraphBlog</u>
 - <u>medium.com/tag/oracle-graph/latest</u>
- Youtube channel
 - youtube.com/c/OracleSpatialandGraph

Find us on social media:

- Twitter: @SpatialHannes, @kpatenge, @ryotaymnk, @AnnamalaiMelli
- LinkedIn: Oracle Spatial and Graph Group













Save the Date!

Analytics and Data Summit 2024

April 9-11, 2024 Oracle Conference Center Redwood Shores, California

www.andouc.org/andsummit2024







