Hands on Lab with Neo4j and Amazon Bedrock

Neo4j and Amazon Bedrock

Introduction to Neo4j
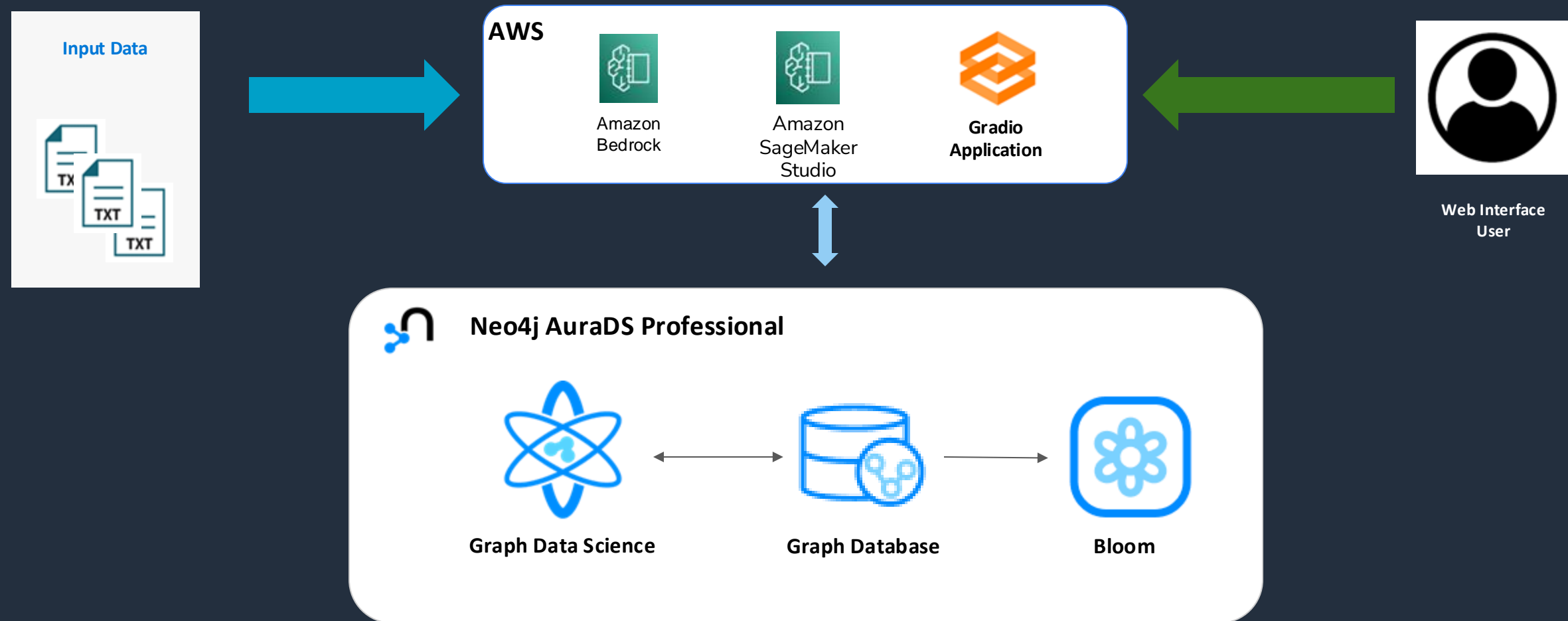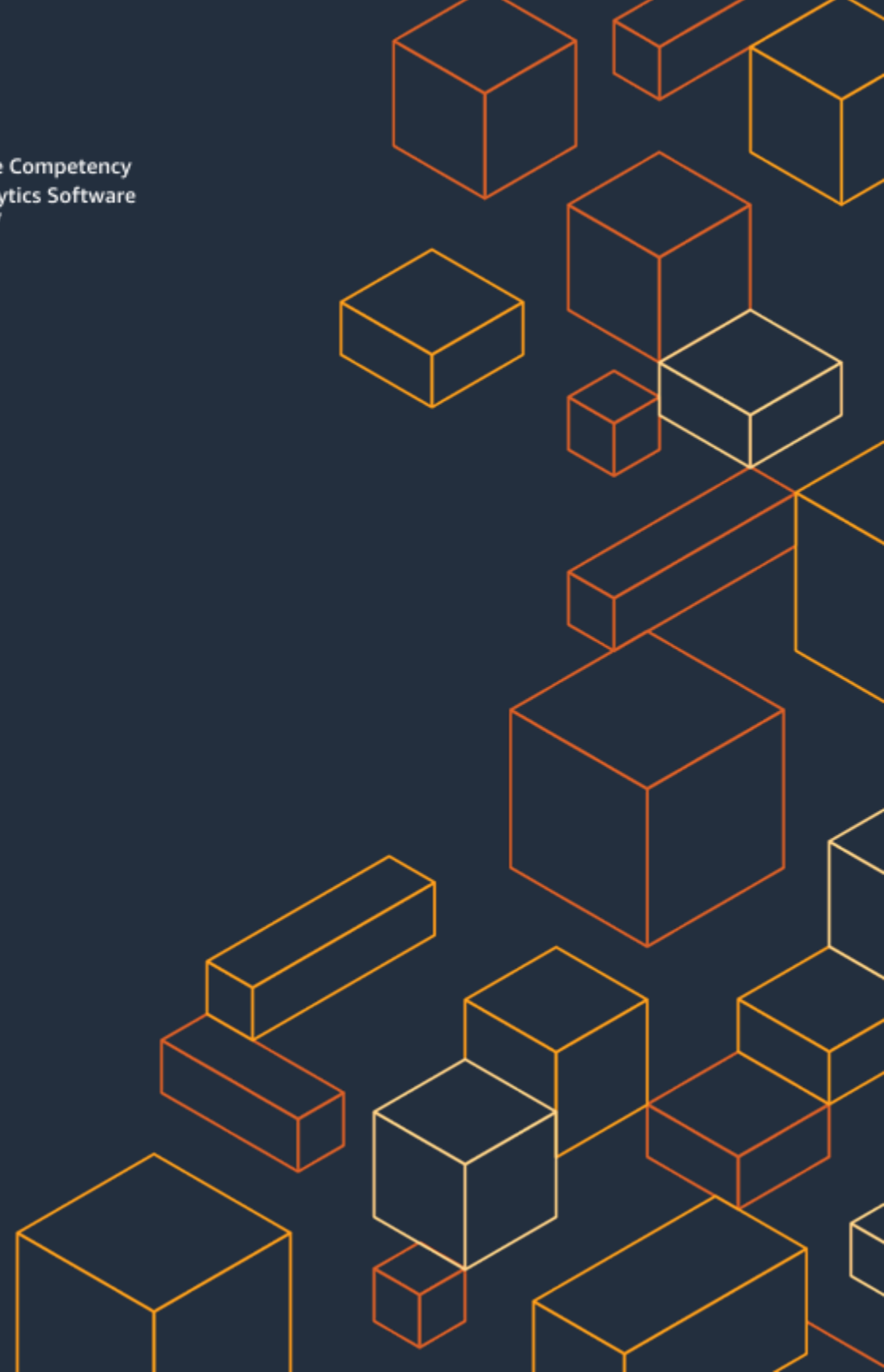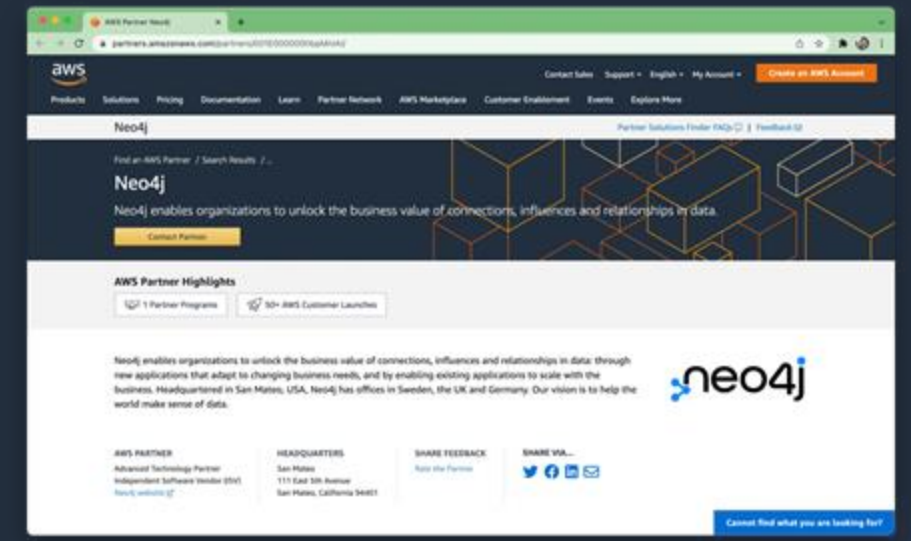
aws

neo4j

aws PARTNER

- Public Sector
- AWS PrivateLink Ready
- Amazon SageMaker Ready
- AWS Marketplace Seller

- ML Software Competency
- Data & Analytics Software Competency

Neo4j and AWS Partnership

- >40% of Neo4j customers run on AWS
- Member of AWS Partner Network since 2013
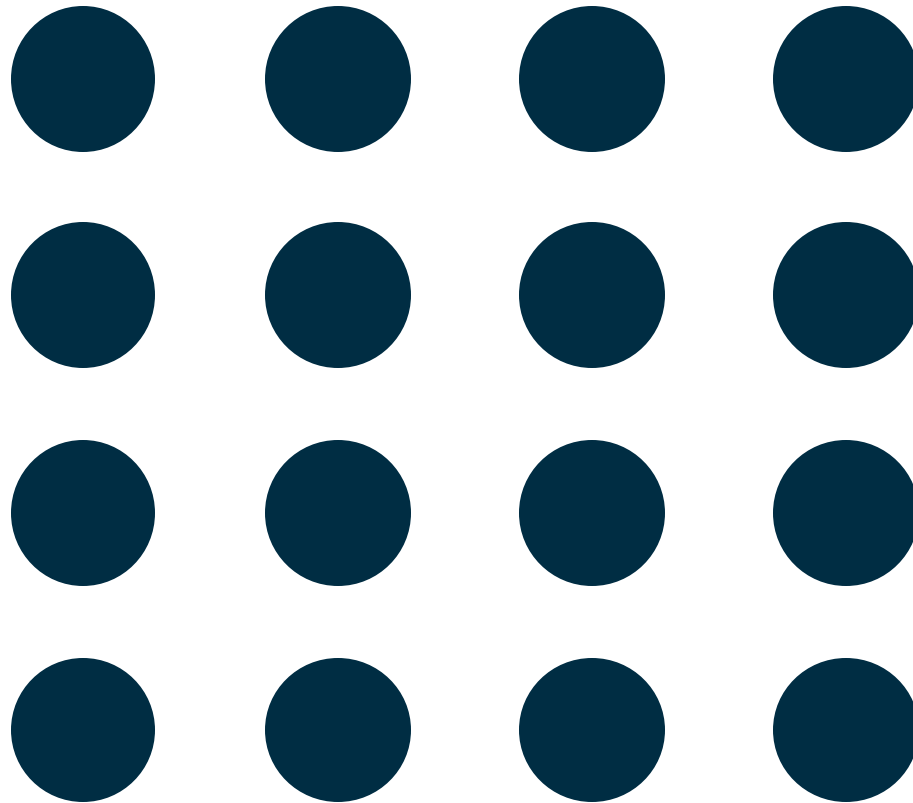- Collaborative Joint Engineering

- APN Advanced Tier Partner
- AWS ISV Workload Migration
- APN Global Startup
- ISV Accelerate
- SRRP

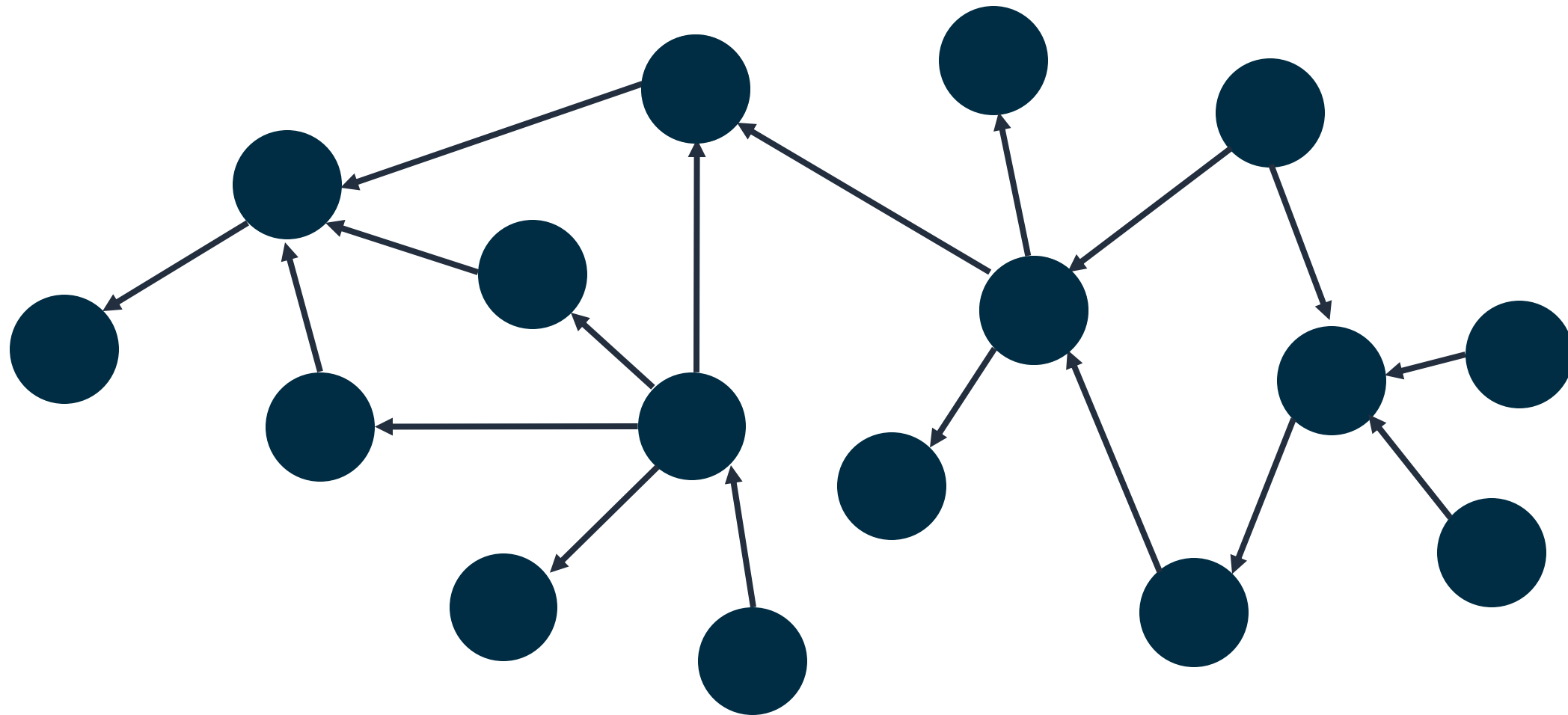- AWS Marketplace Seller
- AWS Public Sector Partner

- ML Competency
- Data and Analytics Competency

- SageMaker Service Ready
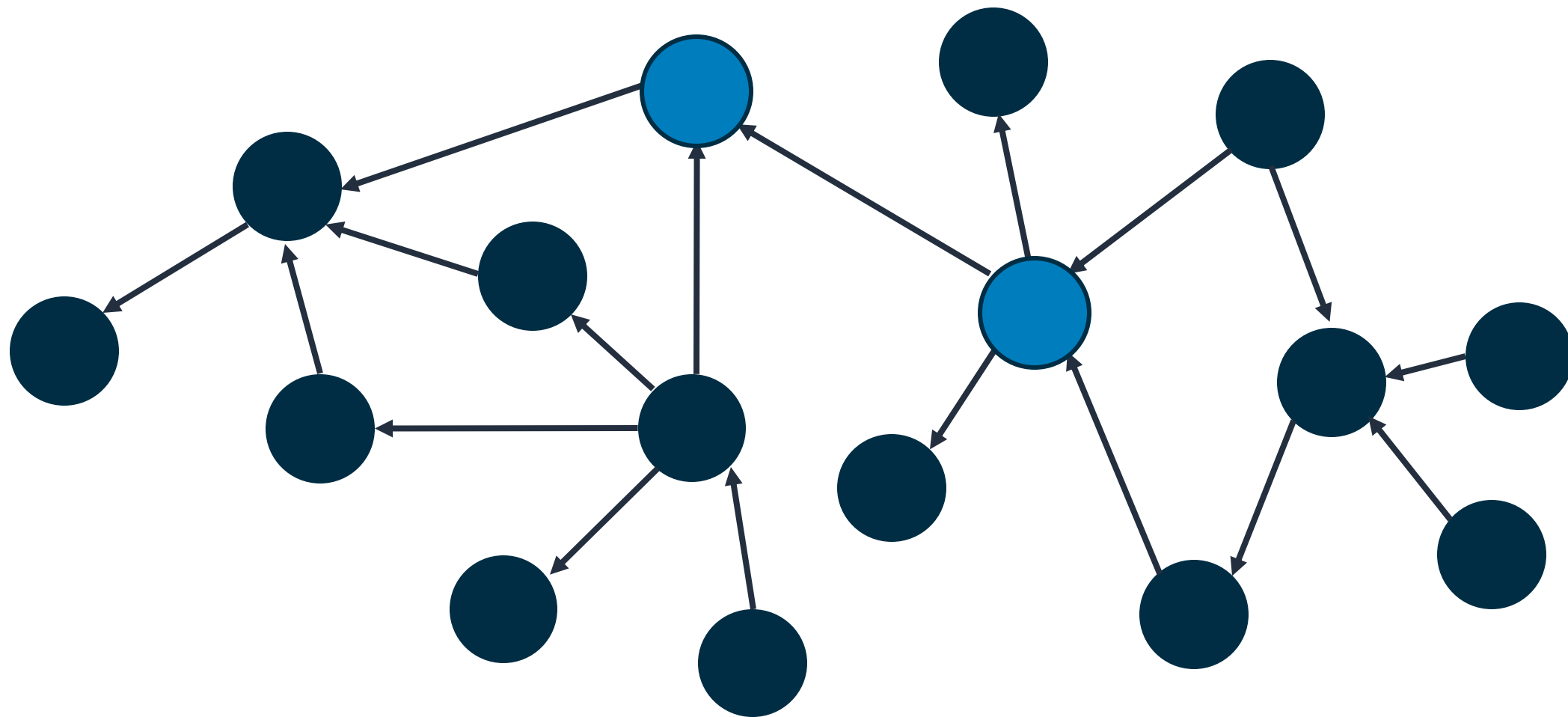- PrivateLink Service Ready

It's Not What You Know

It's Who You Know

It's Who You Know *And Where* They Are

aws

Why use a graph database?

X360°

- Customer 360° **⇐ most common 1st use case**
- Product 360°
- Employee 360°

Enterprise Data Fabric (EDF)

- Data Unification
- Master Data Management

Process

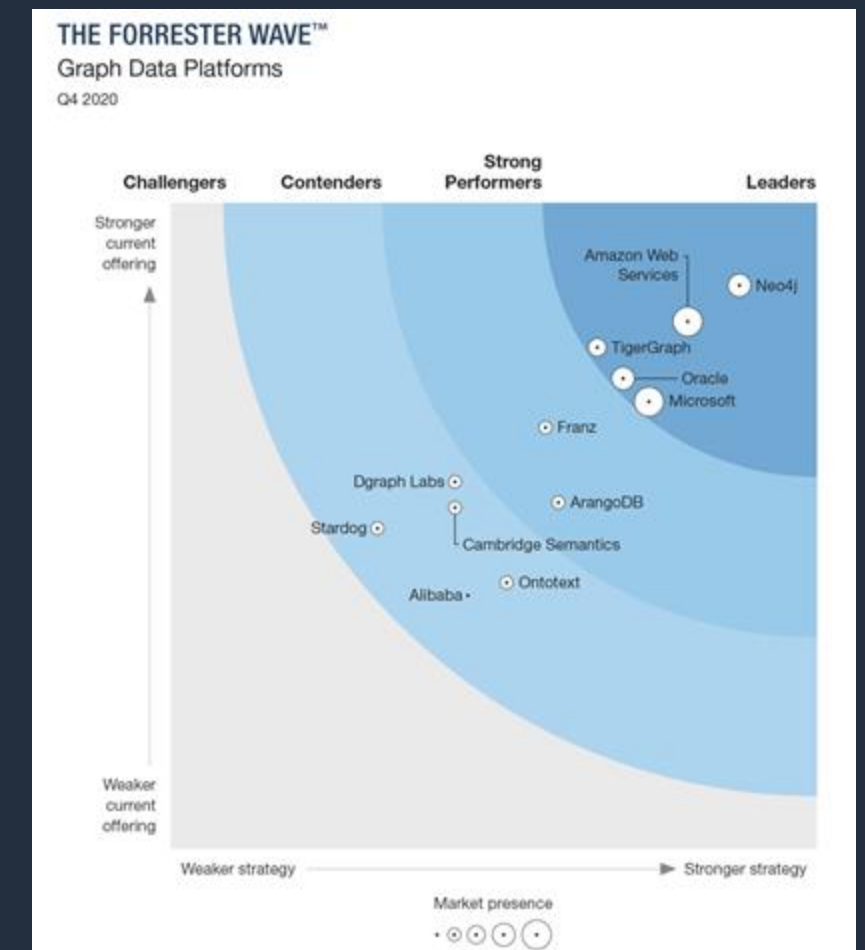- Customer Journey
- Lead to Order
- Order to Cash

Risk

- IT Configuration Management
- Cybersecurity / Access Control
- Supply Chain / Vendor Management
- Internal Audit / Fraud Detection

Neo4j is the leading graph database.

- The first graph data science platform
- The most flexible graph data model
- The easiest-to-use graph query language

THE FORRESTER WAVE™
Graph Data Platforms
Q4 2020

Thousands of organizations use Neo4j.

20 of the top 25 financial firms
7 of the top 10 retailers
7 of the top 10 software vendors

Select Neo4j and AWS Joint Customers

aws

Neo4j is easy to use on AWS

Wait… What is a (Property) Graph?!

Nodes
- Can have Labels to classify nodes
- Labels have native indexes

Relationships
- Relate nodes by type and direction

Properties
- Attributes of Nodes & Relationships
- Stored as Name/Value pairs
- Can have indexes and composite indexes
- Visibility security by user/role

name: "Dan"
born: May 29, 1970
twitter: "@dan"

name: "Ann"
born: Dec 5, 1975
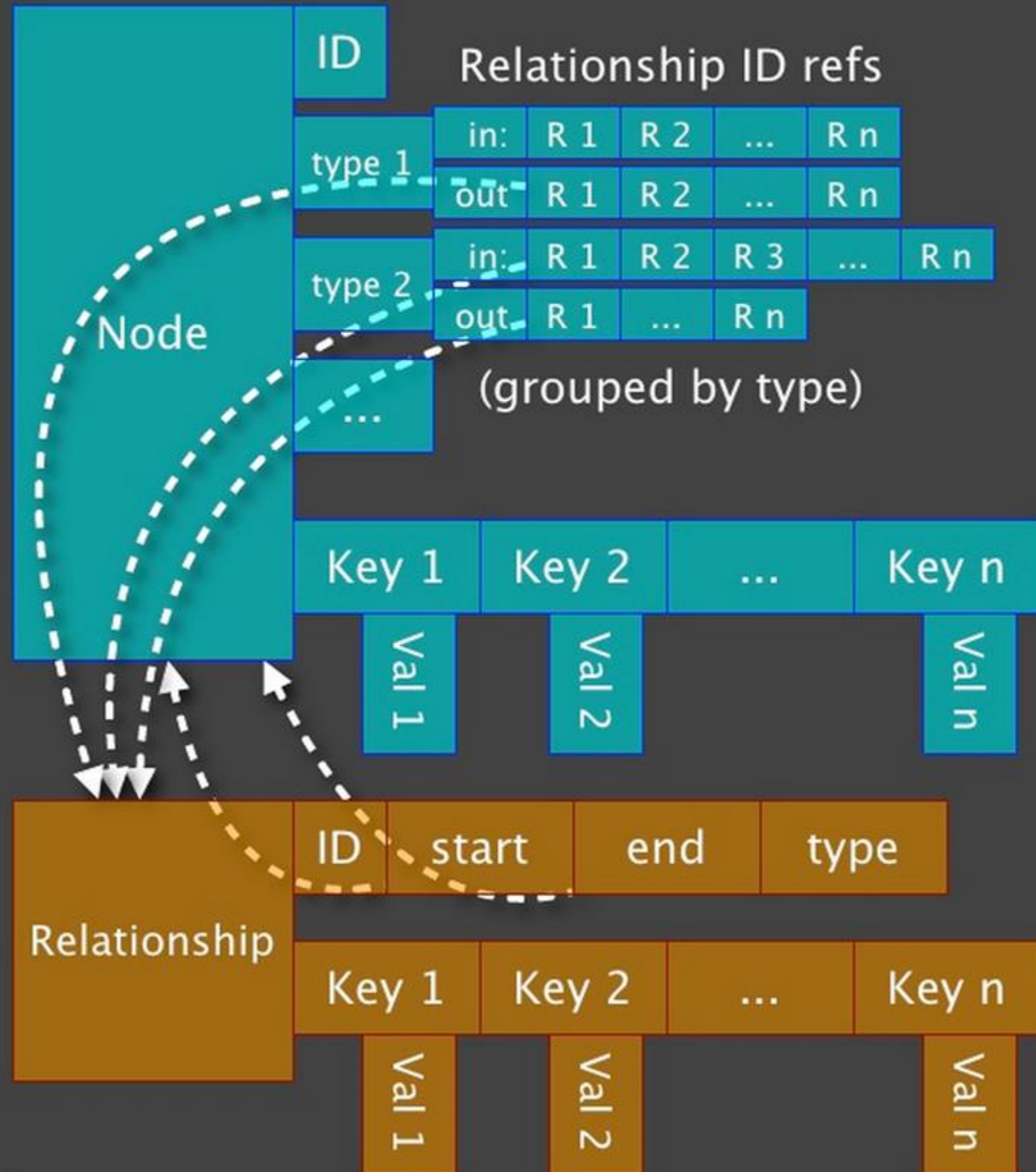
PERSON

MARRIED TO

LOVES

LOVES

PERSON

DRIVES

OWNS

since:
Jan 10, 2011

since:
Dec 14, 2010

CAR

brand: "Volvo"
model: "V70"
Latitude: 37.5629900°
Longitude: -122.3255300°

aws

What we put in cache

Node

| ID | Relationship ID refs |

| type 1 | in: | R 1 | R 2 | ... | R n |
| | out | R 1 | R 2 | ... | R n |

| type 2 | in: | R 1 | R 2 | R 3 | ... | R n |
| | out | R 1 | ... | R n |

(grouped by type)

| Key 1 | Key 2 | ... | Key n |
| Val 1 | Val 2 | | Val n |

Relationship

| ID | start | end | type |

| Key 1 | Key 2 | ... | Key n |
| Val 1 | Val 2 | | Val n |

Neo4j Secret Sauce

1 **Pointers instead of Lookups**

2 **Fixed Sized Records**

3 **"Joins" on Creation**

4 **Spin Spin Spin through this data structure**

aws neo4j

Now, let's get started with the lab…

https://github.com/neo4j-partners
https://github.com/neo4j-partners/hands-on-lab-neo4j-and-bedrock

Complete labs 0, 1 and 2.