Experience the functionality of

Neo4j Graph Database

Performance at Scale

No joins graph traversal consistently delivers performance in milliseconds

Fast K-hop query returns results blaz - ingly fast across multiple hops

Enhanced indexing and query planning

Autonomous Clustering for low-touch horizontal scaleout and efficient use of servers

Scale up with Fabric and query across databases databases or shards

Developer Productivity

Property graph model: what you model is what gets stored in the database

Cypher, a graph query language easier and more powerful and expressive than SQL

Official support for .Net, Java, JavaS - cript, Go, and Python drivers and other community supported ones; GraphQL library for rapid front-end application development

Variety of tools for low-code data modeling and loading (Data Importer), user-friendly graph query explorer (Browser), and no-code data visualiza- tion (Bloom)

A set of connectors for i nteroperability with the rest of your intelligent data ecosystem, including for Apache Kafka, cloud data warehouses, and business intelligence tools

Operational Trust

ACID compliance guarantees data integrity for OLTP/OLAP workloads

SSO and LDAP/directory services inte- gration with your security ecosystem

RBAC for fine-grained security of all nodes, properties, and relationships

Self managed or fully managed no-ops Neo4j Aura

Neo4j Ops Managerfor UI-based opera- tions of all self-managed deployments

DevOps in the cloud with Helm Charts for Kubernetes

Batch import to load large and incre- mental amounts of data

Any-to-any rollingupgrades for anytime upgrades with no downtime

Differential backup and restore, includ- ing restore until a point in time

Unmatched Support and Services

World's foremost graph experts in professional services and technical support

ROI maximization and project prioritiza- tion by customer success managers

Free online graph courses and paid customized trainings

Largest and most active community of graph practitioners

neo4j

The Most Trusted Database for Intelligent Applications

Neo4j Graph Database는 개발자가 데이터의 풍부한 관계를 활용하게 하는 지능형 애플리케이션을 만들고 다른 데이터베이스와 비교할 수 없는 속도로 복잡한 질문에 답할 수 있도록 지원하는 데이터 베이스로 시장을 선도하고 있습니다.

Neo4j Graph Data Platform의 핵심인 Neo4j Graph Database는 생명 과학, 유틸리티, 금융 서비스, 사이버 보안 등 산업 전반에 걸쳐 기업이 실시간으로 숨겨진 인사이트를 찾을 수 있도록 지원합니다.

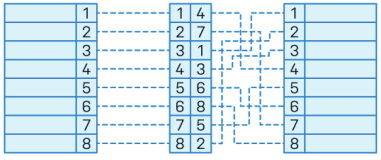
5 Advantages of

Graph Databases vs. Relational Databases

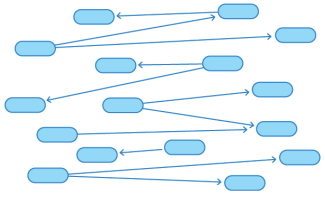
Relational Database	Neo4j Graph Database

1. 빠른 쿼리 속도

Lots of joins make queries more com-plex and slower when they go beyond one level of relationships

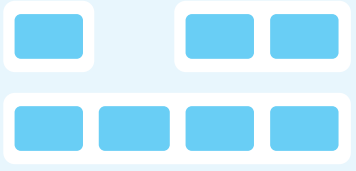


Querying relationships is fast and simple– even when they go beyond one level of relationships.

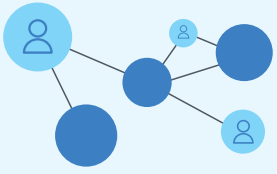


2. 더 깊은 관계 이해

Integrating siloed and distributed datasets is complex and challenging to do in a performant way (e.g., complex ETLs, correlating disparate data, etc.).

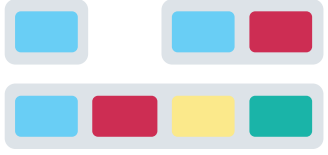


Graphs make it easy to navigatedeep level data relationships. Siloed datasets can easily be brought together in a per-formant way to support deep insights.

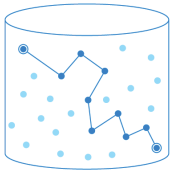


3.확장 유연성

Changing business needs leads to frequent schema changes and applica-tion or database redesign.



Responding to change is easierwith schema flexibility and has minimal impact on query performance.

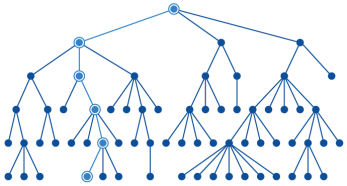


Relational Database	Neo4j Graph Database

4. 연결된 데이터 분석

Relational databases are best suited to access data with simple logical conditions and aggregations.

Graph databases are suited to use connected data to address problems like pattern matching, routing, and massively connected data.



5. 직관적인 쿼리 언어

SQL queries are complex for many recursive traversals and joins. This leads to code that is confusing and difficult to maintain.

```
SQL queries

SELECT name, p2.name as Manager
 FROM Person
LEFT JOIN Person_Department
  ON Person.Id = Person_Department.PersonId
LEFT JOIN Department
  ON Department.Id = Person_Department.DepartmentId
LEFT JOIN Person p2
  ON Person.MgrId = p2.PersonId
WHERE Department.name = "IT Department"
```

Cypher queries are intuitive and easy to understand. This makes them faster to create and maintain and reduces chances of errors.

```
Cypher query

MATCH (mgr:Person)-[:MANAGES]→(p:Person)-[:WORKS_IN]→(d:Department)
WHERE d.name = "IT Department"
RETURN p.name, mgr.name as Manager
```

그래프 데이터베이스를 통해 향상된 데이터 인사이트를 확보하세요!

Data+ AI에 특화된 클라우드 MSP 클루커스는
Neo4j와 파트너십을 통해 고객에게 더 나은 데이터 가치를 제공합니다.

neo4j.com | cloocus.com

neo4j x Cloocus