



# Grafdatatabaser och Cypher

Mats Rydberg [mats@neo4j.org](mailto:mats@neo4j.org)

Linnéa de Llano Balcells [linnea.andersson@neo4j.org](mailto:linnea.andersson@neo4j.org)

# Om mig -- Linnéa de Llano Balcells

- Från Bromölla, bor numera mitt-i-ingenstans (mellan Klippan och Ängelholm)
- Examen från Chalmers 2017
- Arbetar på Neo4j sedan 2020
  - Cypher



# Om mig -- Mats Rydberg

- Bor i Malmö
- Examen från LTH, D-programmet 2014
- Arbetat på Neo4j sedan 2015
  - Cypher, openCypher
  - Grafalgoritmer, GDS



# Grafdatabaser

# Olika typer av databaser

- Relationsdatabas/SQL-databas: *MySQL, PostgreSQL, Oracle*
- NoSQL-databas
  - Nyckel-värde-databas: *Redis*
  - Kolumn-orienterad databas: *Cassandra*
  - Dokumentdatabas: *MongoDB*
  - Grafdatabas: *Neo4j*

# Olika typer av databaser

- Relationsdatabas/SQL-databas: *MySQL, PostgreSQL, Oracle*
- NoSQL-databas
  - Nyckel-värde-databas: *Redis*
  - Kolumn-orienterad databas: *Cassandra*
  - Dokumentdatabas: *MongoDB*
  - Grafdatabas: *Neo4j*

```
> SET mykey "Hello world"  
OK
```

```
> GET mykey  
"Hello world"
```

# Olika typer av databaser

- Relationsdatabas/SQL-databas: *MySQL, PostgreSQL, Oracle*
- NoSQL-databas
  - Nyckel-värde-databas: *Redis*
  - Kolumn-orienterad databas: *Cassandra*
  - Dokumentdatabas: *MongoDB*
  - Grafdatabas: *Neo4j*

```
INSERT INTO people.dancers (id,name,pets)
VALUES(
  123456789,
  'Linnea Andersson',
  { dog : 'Loba', cat : 'Ellie' }
);
```

# Olika typer av databaser

- Relationsdatabas/SQL-databas: *MySQL, PostgreSQL, Oracle*
- NoSQL-databas
  - Nyckel-värde-databas: *Redis*
  - Kolumn-orienterad databas: *Cassandra*
  - Dokumentdatabas: *MongoDB*
  - Grafdatabas: *Neo4j*

```
> db.people.insertOne (
  {
    name: "Linnéa",
    age: 30,
    hobbies: ["swimming", "fishing"],
    partner: { name:"Antón" }
  })

> db.people.find("name": "Linnéa" )
{
  name: "Linnéa",
  age: 30,
  hobbies: ["swimming", "fishing"],
  partner: { name:"Antón" }
}
```



# Olika typer av databaser

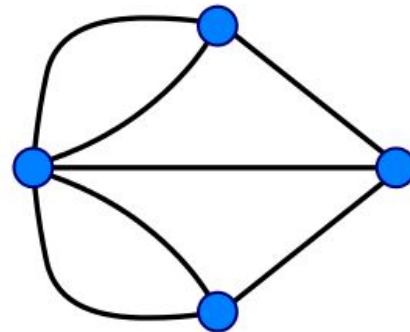
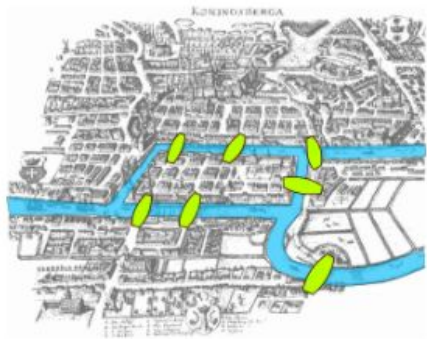
- Relationsdatabas/SQL-databas: *MySQL, PostgreSQL, Oracle*
- NoSQL-databas
  - Nyckel-värde-databas: *Redis*
  - Kolumn-orienterad databas: *Cassandra*
  - Dokumentdatabas: *MongoDB*
  - Grafdatabas: *Neo4j*

# Königsbergs sju broar och grafteori

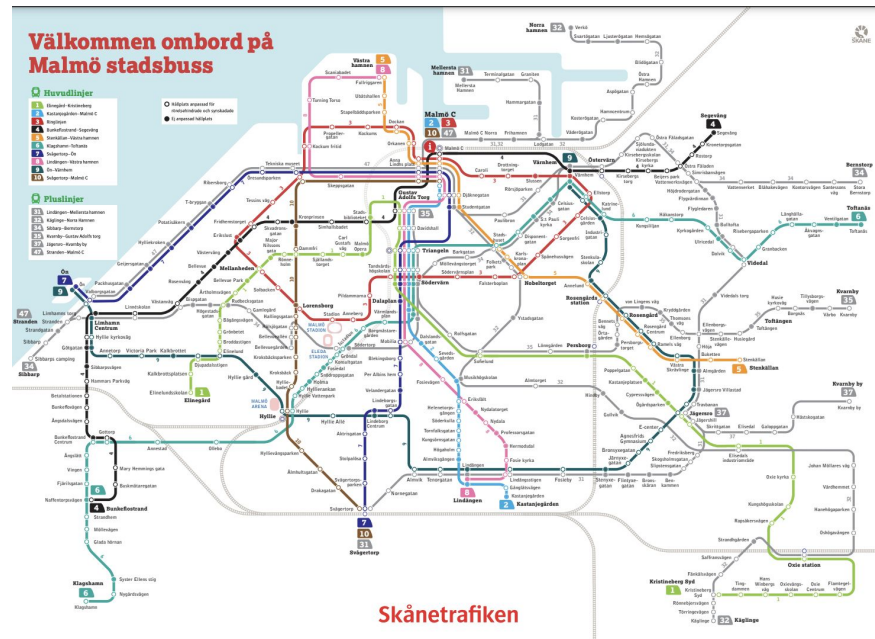
Leonhard Euler



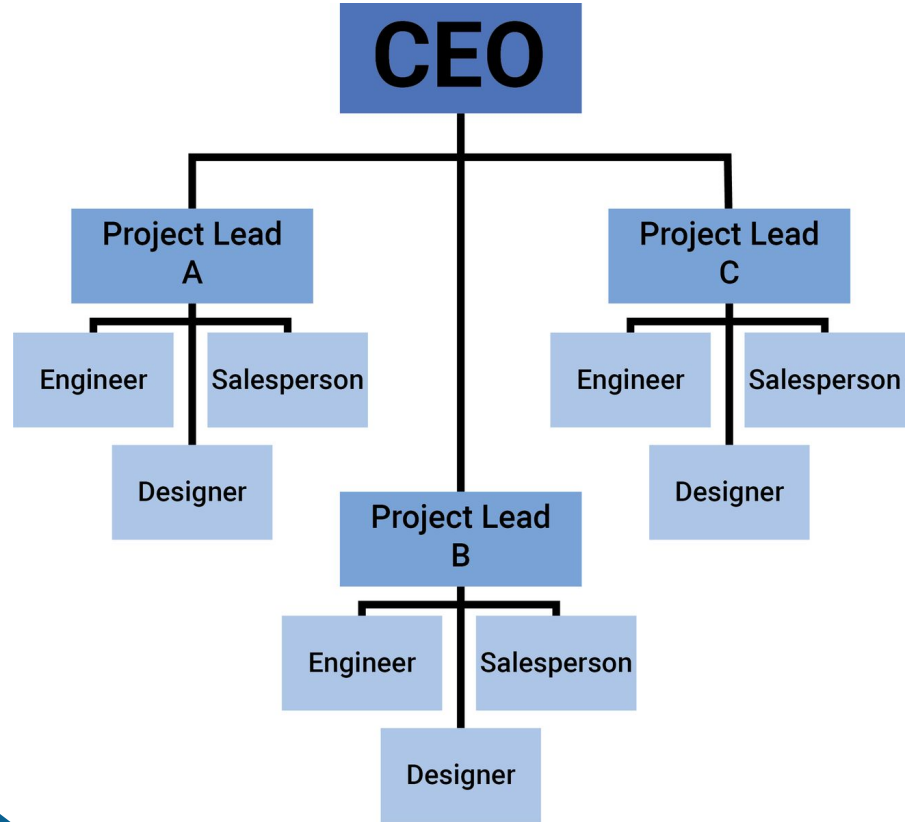
$$G = (V, E)$$



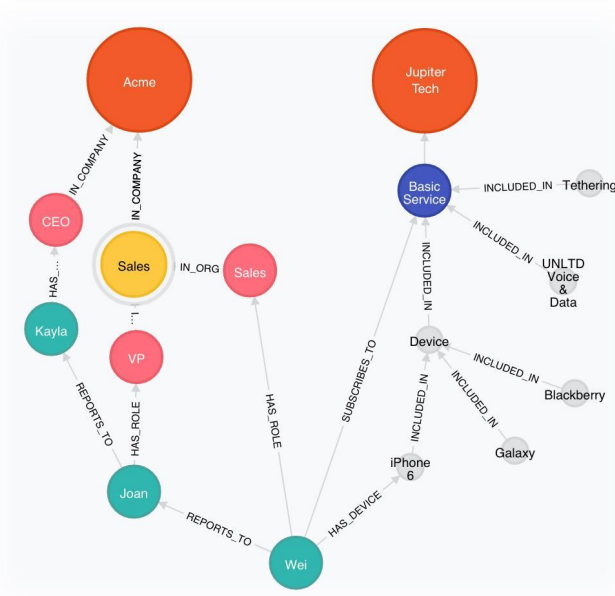
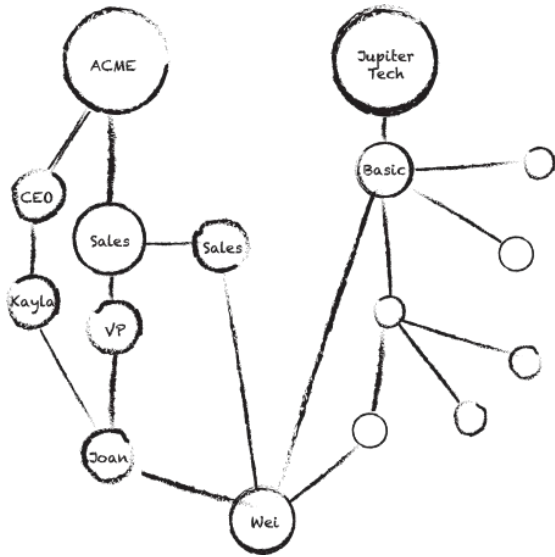
# Kartor



# Organisations graf

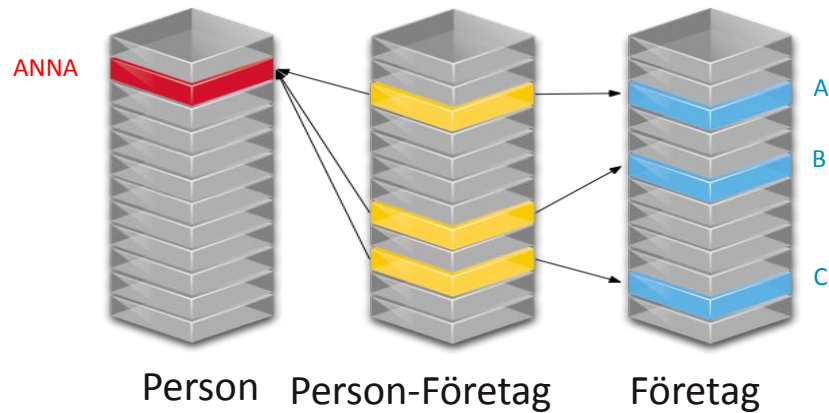


# Whiteboard-modell vs. fysisk modell

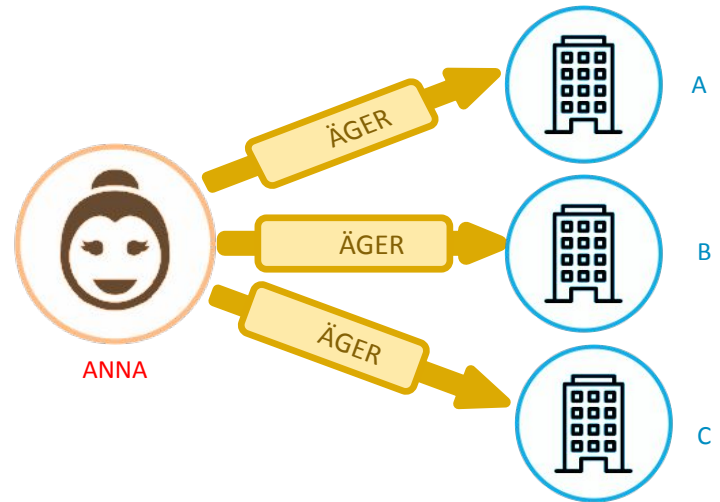


# Relationsdatabas vs. grafdatabas

## Relationsdatabas



## Grafdatabas



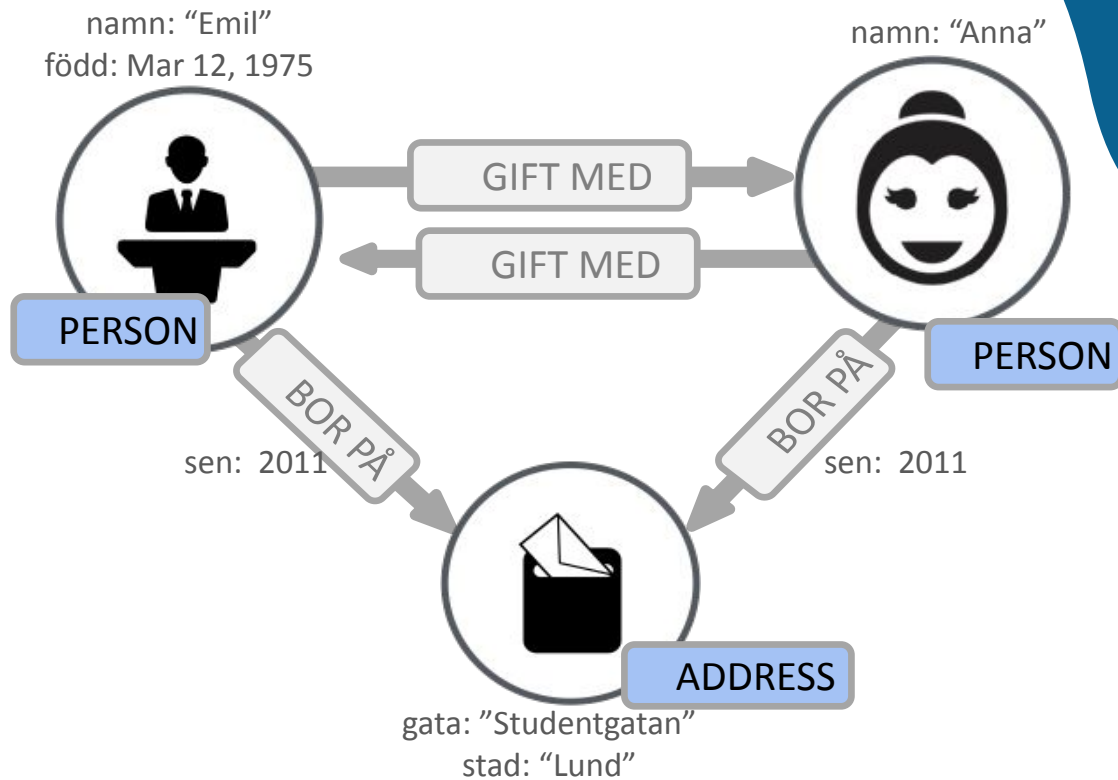
# Noder och relationer

## Noder

- Objekten i grafen
- Kan ha typ (*labels*)
- Kan ha egenskaper (*properties*)

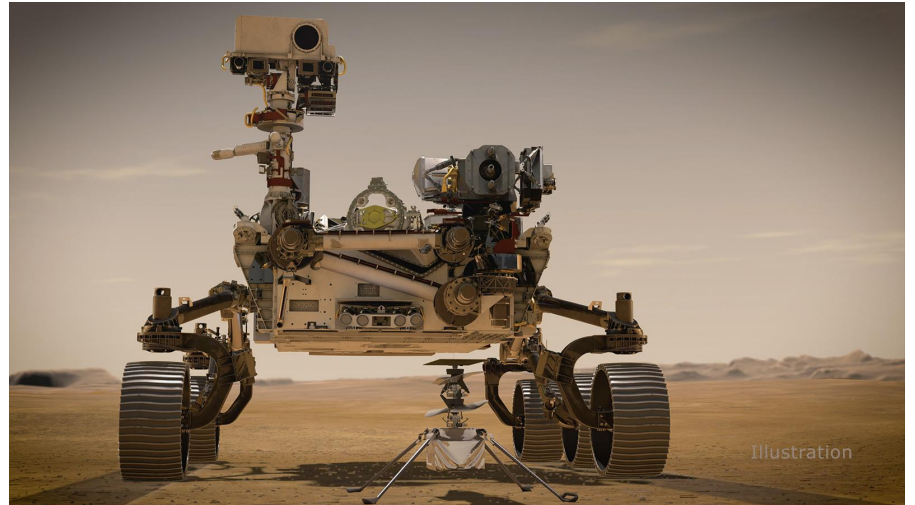
## Relationer

- Binder samman noderna med typ (*type*) och riktning
- Kan ha egenskaper (*properties*)



# Vanliga användningsområden

- Spårning av bedrägerier
- Köprekommendationer
- Kunskapsgrafer
- Nätverk och IT-system





# Use case: Social networks

Use Neo4j to build a social network application or analyze data in a social network.

NBC News used Neo4j to map and analyze the connections between 200,000 tweets by Russian Twitter trolls that interfered in the 2016 U.S. Presidential election

## Use case: Knowledge graphs

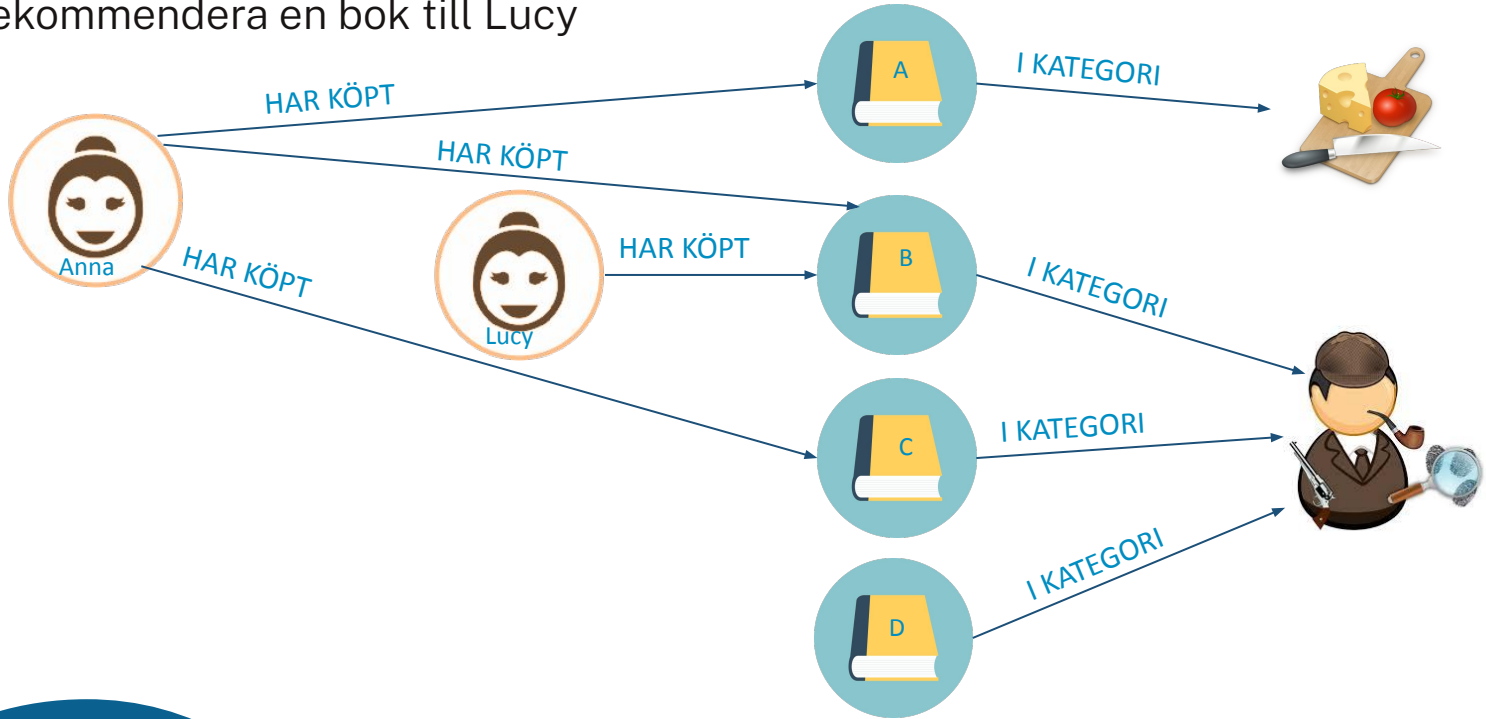
A satellite in space, viewed from a low angle. The satellite has a cylindrical body with a camera lens on the right side and solar panels extending from the top. The background is a dark blue and white streaked pattern, suggesting motion or a data stream.

Use Neo4j to combine data from various sources into one graph and derive more insights from it by using data analytics

NASA is using a Neo4j knowledge graph as a "lessons learned" database to document, categorize and search their collective knowledge from past missions.

# Köprekommendation grafdataabas

- Rekommendera en bok till Lucy



# Köprekommendation relationsdatabas

Användare

id	namn
0	Anna
1	Lucy

Kategori

id	kategori
0	kokbok
1	deckare

Köp

person_id	bok_id
0	0
0	1
0	2
1	1

Bok

id	bok	kategori_id
0	A	0
1	B	1
2	C	1
3	D	1

# Frågespråk & Cypher

# Frågespråk

- Huvudfunktion för att arbeta med databasen
- Uppdatera med ny data
- Formulera analytiska frågor

**SQL** är en standard med flera decenniers historia; ett av de mest använda människa/maskin-språken någonsin.

# Frågespråk, SQL

Leta efter användare som postat mycket på ett forum under den senaste tiden:

```
SELECT users.username, users.joinDate, COUNT(*) AS numberOfPosts
FROM users
INNER JOIN posts ON users.id = posts.userId
INNER JOIN logins ON users.id = logins.userId
WHERE (SELECT MAX(logins.date) AS lastLoginDate) > ?
GROUP BY users.id
ORDER BY numberOfPosts DESC
```

# Frågespråk, SQL

```
SELECT users.username, users.joinDate, COUNT(*) AS numberOfPosts
FROM users
INNER JOIN posts ON users.id = posts.userId
INNER JOIN logins ON users.id = logins.userId
WHERE (SELECT MAX(logins.date) AS lastLoginDate) > ?
GROUP BY users.id
ORDER BY numberOfPosts DESC
```

- Tre tabeller: users, posts och logins
- "slås ihop" (join) för att skapa en tabell med användare, deras poster samt inloggningar
- filtreras med datumfilter på senaste (maximala) inloggningen



# Frågespråk, SQL

```
SELECT users.username, users.joinDate, COUNT(*) AS numberOfPosts
FROM users
INNER JOIN posts ON users.id = posts.userId
INNER JOIN logins ON users.id = logins.userId
WHERE (SELECT MAX(logins.date) AS lastLoginDate) > ?
GROUP BY users.id
ORDER BY numberOfPosts DESC
```

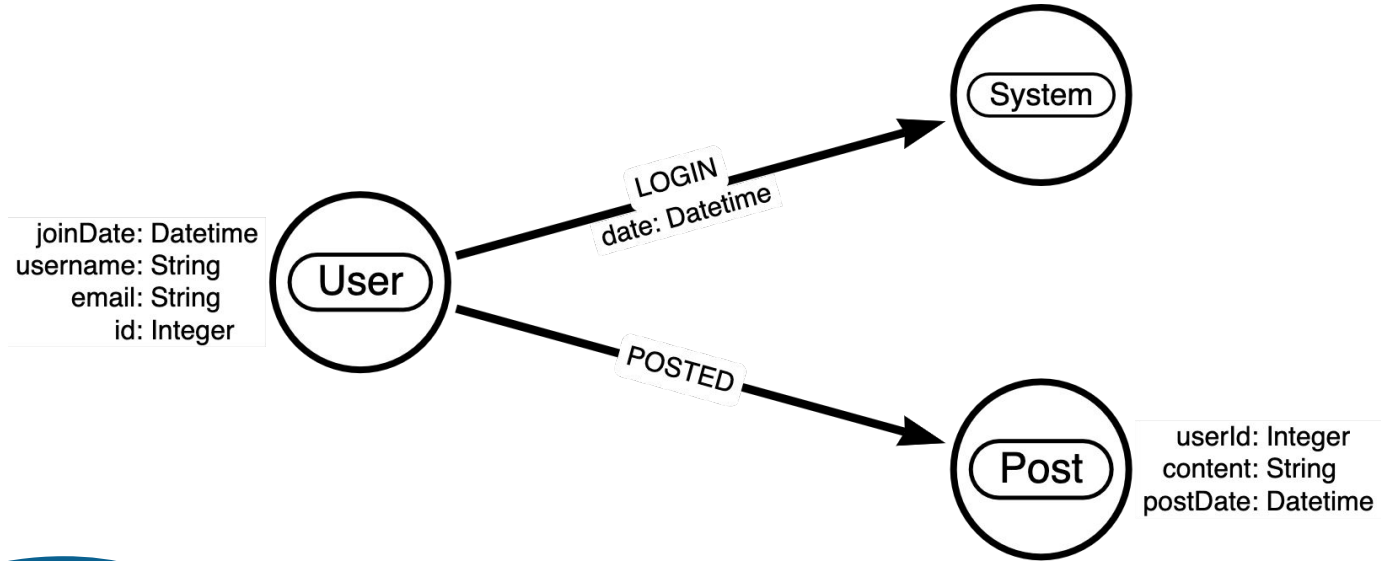
- grupperas på användares id
- aggregeras (räknas) över grupperna
  - dvs, vi räknar antalet poster varje unik användare har gjort

# Frågespråk, SQL

- Detta är ett vanligt mönster i SQL: joina tabeller för att söka mellan datapunkter
- Tabeller representerar vanligen olika datatyper och lagrar data som är *likformig* i närhet
- Optimerat för att effektivt processera stora mängder av likformig data
- Inte optimerat för att hitta *relationer* i databaser
  - Namnet "relationsdatabas" (relational database) till trots!

# Datamodell

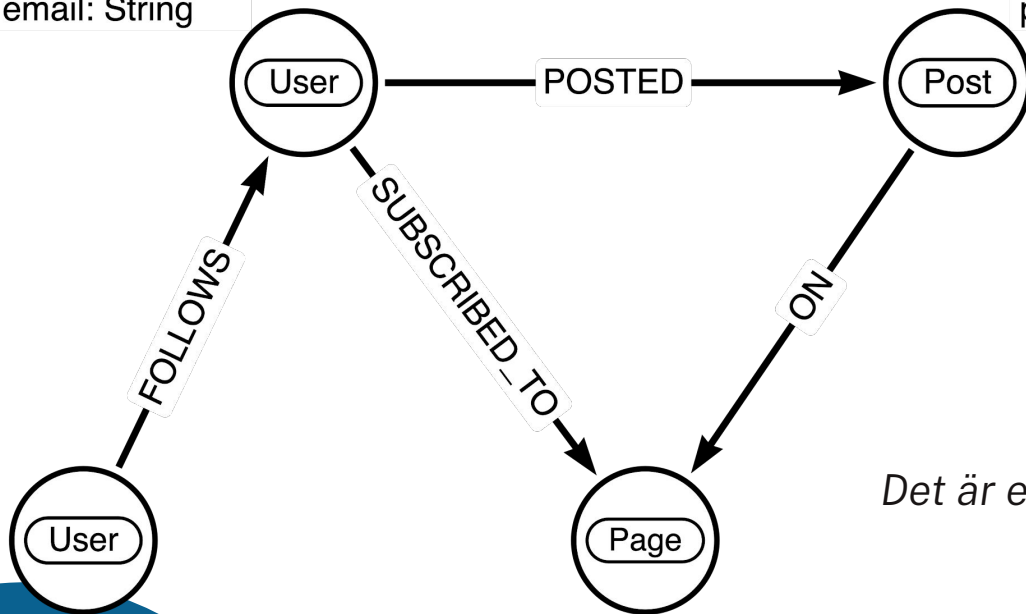
Hur ser egentligen datamodellen här ut?



# Datamodell

joinDate: Datetime  
username: String  
email: String

userId: Integer  
content: String  
postDate: Datetime



*Det är en graf!*

# Datamodell

För att modellera datamodellen som tabeller krävs avancerade protokoll som mappar dataobjekt mellan den logiska modellen och den fysiska modellen i databasen.

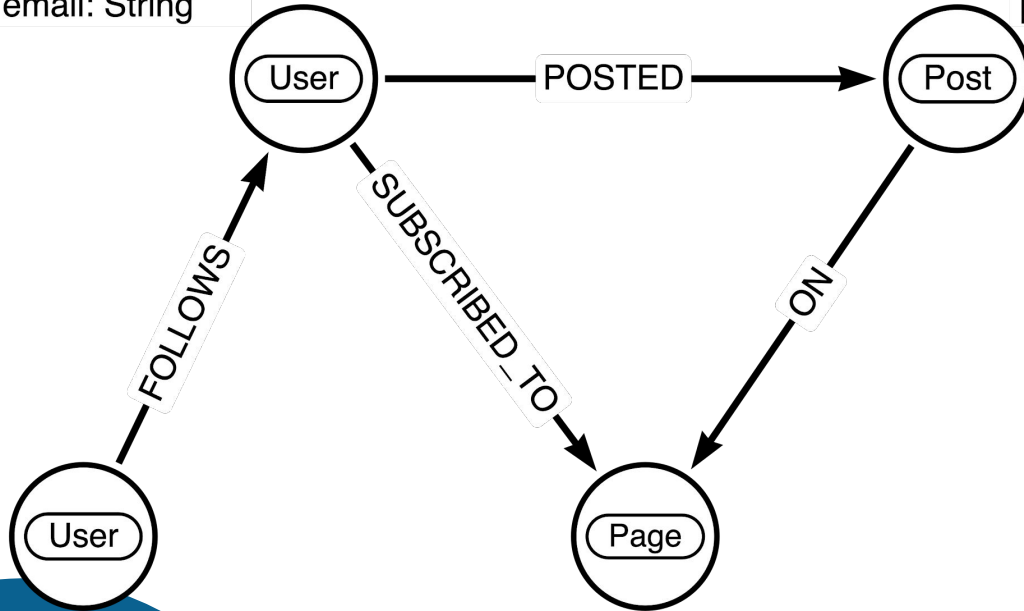
Det skapar ett avstånd mellan hur användaren arbetar med modellen (via frågespråket; SQL) och hur användaren tänker på modellen.

I en grafmodell är detta avstånd frånvarande!

# Datamodell

joinDate: Datetime  
username: String  
email: String

userId: Integer  
content: String  
postDate: Datetime



# Cypher

- Baserad på mönstermatchning (pattern matching)
- Modellen i frågespråket är samma som datamodellen
  - Ställ en fråga till en graf som en (del)graf
- Mönster uttryckts med s.k. Ascii art
  - ()
  - ()-->()
  - ()<--()--(), ()-->()

# Cypher, nodmönster

`(n)`

- Nod med variabel

`(:Person)`

- Nod med typ ("label"), utan variabel

`{name: 'Mats'}`

- Nod med egenskap ("property"), utan variabel

`(n:Person {name: 'Mats', birthYear: 1988})`

- Sammanslaget av ovan



# Cypher, relationsmönster

`()<--()`

- Riktad relation

`()--()`

- Oriktad relation, alla riktningar matchar

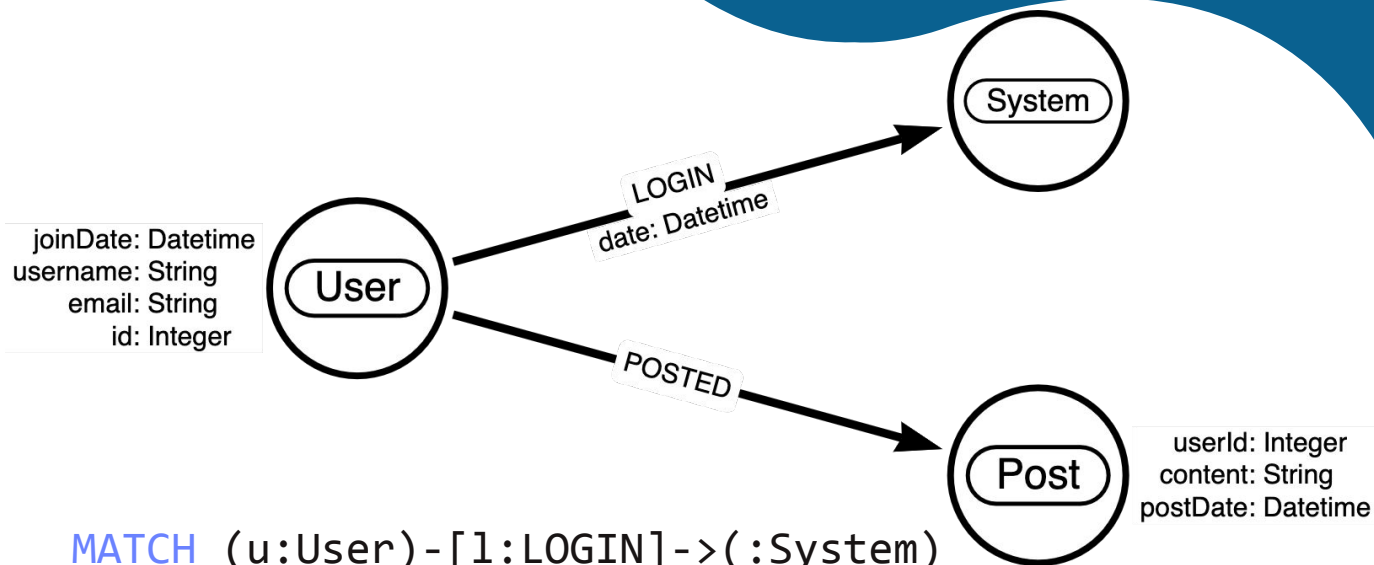
`()-[r:KNOWS]->()`

- Relation med typ och variabel

`()-[:KNOWS {since: '2019'}]->()`

- Relation med typ och egenskap

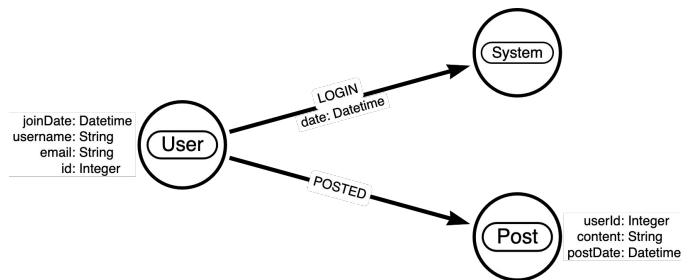
# Cypher



```
MATCH (u:User)-[1:LOGIN]->(System)
WITH u, max(1.date) AS mostRecentLogin
WHERE mostRecentLogin > $lastLogin
MATCH (u:User)-[:POSTED]->(p:Post)
WITH u, count(p) AS numberOfPosts
RETURN u.joinDate, u.username, numberOfPosts
ORDER BY numberOfPosts DESC
```

# Cypher

```
MATCH (u:User)-[l:LOGIN]->(s:System)
WITH u, max(l.date) AS mostRecentLogin
WHERE mostRecentLogin > $lastLogin
MATCH (u:User)-[p:POSTED]->(p:Post)
WITH u, count(p) AS numberOfPosts
RETURN u.joinDate, u.username, numberOfPosts
ORDER BY numberOfPosts DESC
```



# Cypher, summerat

- Datamodellen är samma som whiteboard
- Även språkmodellen är samma
- Den konceptuella modellen behöver inte översättas till olika format under utvecklingen från idé till implementation

Läs mer om [Cypher i manualen](#).

# PAUS

Vi svarar på era frågor!  
Glöm inte plocka med er lite swag :)

# Neo4j, företaget

# Neo4j, företaget

- Grundat 2007 i Lund
- Kommersiellt huvudkontor i San Mateo, Silicon Valley, CA
- Ingenjörshuvudkontor i Malmö
  - 108 ingenjörer
  - 229 totalt
    - i London eller via distans
    - många har examen från LTH
- ~750 anställda totalt



# Att jobba på Neo4j

- Flexibla arbetstider
- Självorganiserande arbetslag
  - 2-12 personer
  - Självbestämda processer
- Smarta, drivna och hjälpsamma medarbetare
- Sociala aktiviteter såsom
  - Quiz, AW, bio, konferenser, digitala fika, m.m.
  - lunchföreläsningar
  - kontorsevent efter säsong







### SF Bay Area

Neo4j, Inc.  
400 Concar Dr.  
San Mateo, CA 94402, USA



### Malmö

Nordenskiöldsgatan 24  
211 19 Malmö,  
6th floor, Sweden



### London

Union House  
182-194 Union Street  
London, SE1 0LH, UK



### Germany

Viktualienmarkt 8  
c/o WorkRepublic  
80331 München, Germany

<https://neo4j.com/careers/>

Location	Department
EMEA	Engineering

Full Stack Engineer Neo4j Aura	Console - Aura	Malmö
Full Stack Engineer Neo4j Aura	Console - Aura	London
Software Engineer - Aura Platform	Platform - Aura	London
Software Engineer - Aura Platform	Platform - Aura	Malmö
Software Engineer - Aura, Usage & Billing Systems	Platform - Aura	Malmö
Software Engineer - Aura, Usage & Billing Systems	Platform - Aura	London
Software Engineer - Cypher Planner	Cypher - Planner	Malmö
Software Engineer - Cypher Planner	Cypher - Planner	London
Software Engineer Orchestration	Orchestration - Aura	London
Software Engineer Orchestration	Orchestration - Aura	Malmö
Summer 2024 internship - Design system (React/Typescript)	Design System	Malmö
Thesis Topic Opportunity (Fall 2024)	Kernel	Malmö

**Vi rekryterar!  
Exjobb!**

# Exjobb på Neo4j

- Grafer, grafer, grafer
- Men också
  - parallella system
  - språkdesign
  - benchmarking
  - molnarkitektur
  - grafalgoritmer
  - databasarkitektur
  - maskininlärning
- [Sommaren 2024](#)
  - Intresseanmälan i det fall vi lägger upp fler exjobbplatser
- [Hösten 2024](#)
  - Intresseanmälan för höstens "intagning"

<https://neo4j.com/careers/>

# Övning

Utforska en filmdatabas

# Övning

- Låt oss utforska en filmdatabas
- Ni ska få utforska en Neo4j sandbox
  - Den är gratis!
  - Ni behöver en e-postadress eller Google-konto
- Öppna följande adress i er webbläsare:

<https://neo4j.com/sandbox/>



# Övning

The image shows a screenshot of the Neo4j website homepage. The navigation bar at the top includes links for 'Aura Login', 'Partners', 'Company', and 'Support', along with a search icon. The main header features the 'neo4j' logo and navigation links for 'Products', 'Use Cases', 'Developers & Data Scientists', 'Pricing', and 'Learn'. Two prominent buttons are visible: 'Contact Us' and 'Get Started Free'. The main content area has a dark blue background with the headline 'Get started with Neo4j Sandbox while your coffee is still brewing'. Below this, a paragraph states: 'Neo4j is a native graph database, purpose-built to leverage data relationships and enable richer, more intelligent applications'. A blue button labeled 'Launch the Free Sandbox' is circled in red. To the right, a screenshot of the Neo4j Sandbox interface is shown, displaying a table of projects and a sidebar with 'Learn', 'Resources', and 'Support' sections.

Aura Login Partners Company Support

neo4j Products Use Cases Developers & Data Scientists Pricing Learn

Contact Us Get Started Free

## Get started with Neo4j Sandbox while your coffee is still brewing

Neo4j is a native graph database, purpose-built to leverage data relationships and enable richer, more intelligent applications

Launch the Free Sandbox

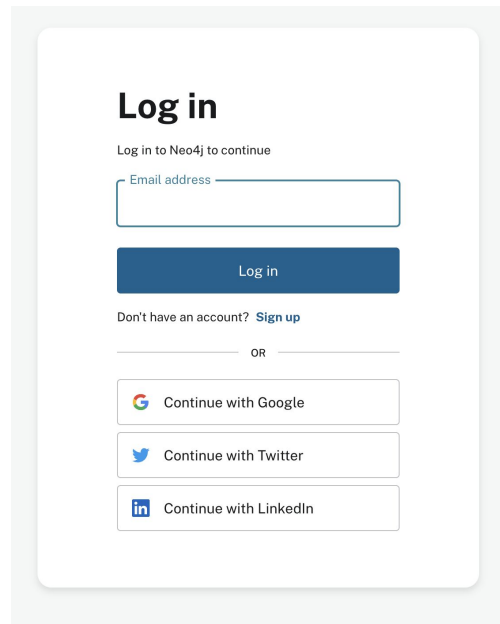
Name	Status
Recommendations	Learning Expires in about 2 days <a href="#">Ping us the details</a>

Learn Resources Support

Done learning? Start developing! [Download Neo4j Sandbox](#)

# Övning

- Ange e-postadress och välj ett säkert lösenord.  
Följ sedan stegen för att registrera dig.
- Eller registrera med Google-konto



The image shows a login interface for Neo4j. At the top, it says "Log in". Below that, it says "Log in to Neo4j to continue". There is a text input field labeled "Email address". Below the input field is a blue button labeled "Log in". Underneath the button, it says "Don't have an account? Sign up". Below this is a horizontal line with "OR" in the center. There are three buttons for social login: "Continue with Google" (with the Google logo), "Continue with Twitter" (with the Twitter logo), and "Continue with LinkedIn" (with the LinkedIn logo).


# Övning

- Välj sedan “Movies” och tryck på “create” (i nedre vänstra hörnet)

Select a project ✕


For Developers (14)  For Data Scientists (7)

Featured Dataset

 Beginner For Developers

**Movies**


A guide to common graph query patterns involving connections between movies, actors, and directors.

 For Developers

**OpenStreetMap**

A graph solution to the shortest-path problem with Cypher involving points of interest and routing of Central Park in New York City.


Libraries Enabled: GraphQL

 Beginner For Data Scientists

**Graph Data Science**

Leverage Neo4j Graph Data Science library to explore graph algorithms for analytics and feature engineering.

Libraries Enabled: Graph Data Science

 For Developers New

**ICIJ OffshoreLeaks**

The Offshore leaks dataset and guide from the International Consortium of Investigative Journalists (ICIJ).

Project : **Movies**

Create

# Övning

- Tryck sedan på “Open”



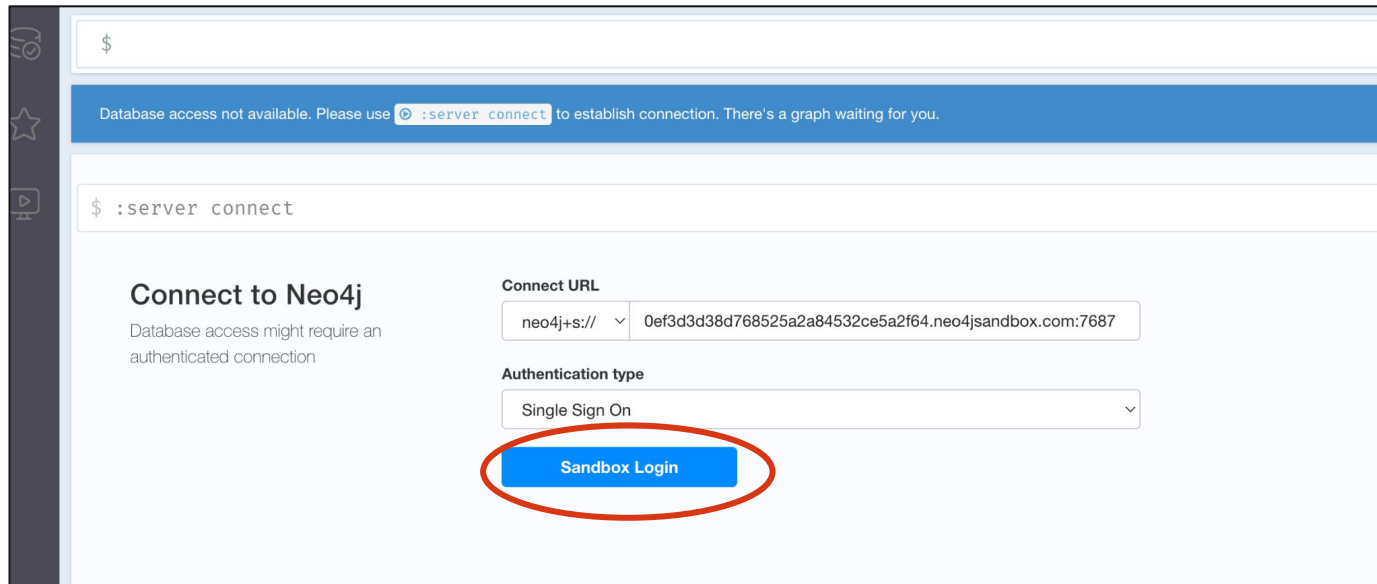
+ New Project

Name	Status	
Movies	Running Expires in about 3 days	<div>Open with Browser</div> <div>Open ▾</div>



# Övning

- Tryck sedan på “Sandbox Login” och logga in med ditt konto



The screenshot shows the Neo4j web interface. At the top, a blue banner reads: "Database access not available. Please use `:server connect` to establish connection. There's a graph waiting for you." Below this, the command `$ :server connect` is entered in the terminal area. The main section is titled "Connect to Neo4j" and contains the following fields:

- Connect URL:** A dropdown menu set to "neo4j+s://" and a text input field containing "0ef3d3d38d768525a2a84532ce5a2f64.neo4jsandbox.com:7687".
- Authentication type:** A dropdown menu set to "Single Sign On".
- Sandbox Login:** A blue button, which is circled in red in the image.

# Övning

- Nu är du redo att börja utforska en filmdatabas!
- Lycka till!

The screenshot shows the Neo4j Cypher interface. At the top, there is a navigation bar with a back arrow and the text "What is Cypher?". Below this, there is a star icon and the title "What is Cypher?". A play button icon is visible on the left side of the main content area. The main content area contains the following text:

Cypher is a graph query language that is used to query the Neo4j Database. Just like you use SQL to query a MySQL database, you would use Cypher to query the Neo4j Database.

A simple Cypher query can look something like this

This content is only revealed when the user clicks the block title.

```
MATCH (m:Movie)
WHERE m.released > 2000
RETURN m LIMIT 5
```

Hint: You can click on the query above to populate it in the editor.

**Expected Result:** The above query will return all the movies that were released after the year 2000 limiting the result to 5 items.

The results are displayed as a graph with five orange circular nodes, each containing the name of a movie: "Cloud Atlas", "Rescue Dawn", "The Matrix Revolutions", "The Da Vinci Code", and "The Matrix Reloaded".

# Övning - Problem

- Hur gammal är personen med namnet **Ian McKellen** (Tips: MATCH)
- Vilken person har regisserat **Joe Versus the Volcano** ?
  - Tips: relationsmönster ()-->()
- Vilken är den äldsta filmen i databasen? (Tips: ORDER BY)
- Vilken film har flest skådespelare? (Tips: count())
- Vilka personer har regisserat flest filmer i vilka någon skådespelare också har regisserat en (annan) film?

<https://neo4j.com/docs/cypher-cheat-sheet/5/neo4j-enterprise/>

# Testa mer på egen hand

- Skapa en Aura free databas: <https://neo4j.com/cloud/aura/>
  - Du kan använda den hur länge du vill!
- Du kan också kolla in andra sandbox-grafer med förladdad data från bl.a:
  - Cybersecurity
  - Fraud detection
  - Twitter
- Och varför inte testa Graph Data Science för att köra olika algoritmer på hela ditt dataset.



Frågor?



# Mer Neo4j!

- Ladda ned [Neo4j Desktop](#) och använd gratis!
- Skapa din egen databas med [Neo4j Aura](#)
- Skapa din egen [Neo4j Sandbox](#)
- Lär dig mer om [Cypher](#)
- Kontakta oss via [Neo4j Community Forum](#)
- Följ (eller medverka till) utvecklingen på [GitHub](#)