Onderzoekscasus DMDD: SQL Server vs Neo4J

20220421 MC de Jonge, Paksha Thullner
20230312 Edit D. Romeijn

Inleiding

Bij deze opdracht onderzoeken we de voor- en nadelen van verschillende databasesystemen bij het implementeren van datamodellen.

We beginnen met het verkennen van Neo4J, de NoSQL database die we bij dit onderzoek als alternatief voor MS SQL Server bekijken. Er volgt een kleine literatuurstudie en daarna een dubbele implementatie binnen een SQL Server en een Neo4J omgeving, met uiteindelijk dezelfde feittypen, constraints en populatie van feiten.

We gebruiken de implementaties om de systemen te vergelijken. Een belangrijk aspect is hierbij de datakwaliteit. De twee databases moeten ook niet-standaard integriteitsregels (mandatory child references of business rules, waarvoor we normaalgesproken stored procedures of iets dergelijks toepassen) bewaken.

Op OnderwijsOnline staat een uitgewerkt conceptueel datamodel. Hiervan is met PowerDesigner een relationele databasestructuur gegenereerd. De code voor de database in Neo4j moet zelf worden uitgewerkt.

Je bouwt de Neo4j-database met alle bijbehorende constraints, vult beide databases met dezelfde testgegevens en laat zien dat ze dezelfde beperkingen hanteren. Mogelijkerwijze zitten er constraints bij die door Neo4J niet te bewaken zijn. De twee implementaties zijn nu goed met elkaar te vergelijken, ook qua datakwaliteit, intension en extension (zie PP Datakwaliteit).

Vervolgens benoem je de voor- en nadelen van de ene implementatie in vergelijking met de andere. Je kan hiervoor zelf criteria opstellen en voor elk criterium onderbouwde argumenten en/of metingen inbrengen.

De bevindingen leg je vast in een onderzoeksverslag.

Deze opdracht voer je met zijn tweeën uit. De verslagen lever je individueel op ISAS in. De deadlines staan in het examenrooster.

Deadlines: Zie het toetsrooster.

Taak 1: Zelfstudie Neo4J

Op #OnderwijsOnline staat een uitgebreide tutorial voor Neo4j. Volg de stappen in deze zelfstudie om vertrouwd te raken met de manier waarop Neo4j werkt.

Taak 2: Stel onderzoeksvragen op

De hoofdvraag van je onderzoek is gegeven: *wat zijn de voor- en nadelen van Neo4J ten opzichte van SQL Server?* De deelvragen moet je echter zelf formuleren. Op welke vragen moet je antwoord hebben voordat je de hoofdvraag kan beantwoorden? Hoe kun je de voorbeelddatabase uit de casus gebruiken om antwoorden te vinden op deze deelvragen? Let erop dat je deelvragen ingaan op kwaliteitsaspecten (daar is het natuurlijk uiteindelijk allemaal om te doen).

Vraag je docent om feedback op je deelvragen.

Taak 3: Literatuuronderzoek

Voer een **klein** literatuuronderzoek uit naar de voor- en nadelen van Neo4J ten opzichte van een relationele database zoals SQL Server. Verken de literatuur over dit onderwerp (boeken, papers, HAN- studiecentrum, internet, ...), en kies minstens twee gedegen bronnen over de voor- en nadelen van Graph-databases in het algemeen en Neo4j in het bijzonder.
Vat de belangrijkste punten uit deze bronnen samen en benoem eventuele verschillen van mening. Noteer je bevindingen en verwerk ze in je onderzoeksrapport. Doe dit in een hoofdstuk **Theoretisch Kader**. Je kunt naar de bevindingen uit dit theoretisch kader verwijzen in de hoofdstukken **Methoden en materialen, Resultaten** en **Discussie** (sloten de bevindingen uit de literatuur aan bij je eigen bevindingen? Waarom wel / niet?).

Taak 4: Maak twee equivalente implementaties

Het CDM (beschikbaar op #OnderwijsOnline) bevat een correct conceptueel informatiemodel in ERM. Uit dit CDM zijn een PDM en een DDL-script voor de SQL-server gegenereerd. De code voor het implementeren van het model als Neo4j- database moeten jullie zelf schrijven.

Let op: Je schrijft zelf een script dat de juiste code genereert. Gebruik vooral géén tools om gegevens uit SQL server te exporteren naar CSV en / of tools om gegevens uit CSV-bestanden te importeren in Neo4J. De ervaring van je voorgangers leert dat je daar veel meer tijd mee kwijt bent dan wanneer je met behulp van code-generatie vanuit SQL Server statements aanmaakt in Cypher, de taal van Neo4J.

De database moet voorbeeldgegevens voor alle entiteiten en attributen bevatten.
Als een constraint te moeilijk is om goed bij één van de twee databases te implementeren, kan je je beperken tot het opleveren van een negatieve SELECT-query (of een soortgelijke Neo4J oplossing) om eventuele gegevensvervuiling op te sporen. Vraag de docent hiervoor om toestemming.

De twee implementaties moeten zich identiek gedragen bij alle mogelijke updatepogingen. De volgende twee punten gelden voor beide implementaties (IMP1 en IMP2):

1. Ze bevatten altijd dezelfde populatie van concrete feiten.
2. Ze handhaven dezelfde business rules.

Dit komt neer op:

- Alle gegevens die in IMP1 zijn toegestaan kunnen ook in IMP2 worden opgeslagen en vice versa.
- Alle gegevens die niet in IMP1 zijn toegestaan zijn dat ook niet in IMP2 en omgekeerd.

Aanbevolen aanpak:

1. Maak een lege database aan in SQL-server en voer het meegeleverde DDL-script uit.
2. Vul de lege tabellen met gegevens die voldoen aan alle integriteitsregels, minimaal 5 records per entiteitstype. Alle verderop genoemde informatiebehoeften moeten met de ingevulde voorbeeldgegevens te beantwoorden zijn, resultaten beschikbaar hebben.
3. Zorg ervoor dat alle integriteitsregels (inclusief mandatory child references) bij de SQL-implementatie worden afgedwongen.
4. Toon aan dat de SQL-implementatie alle integriteitsregels handhaaft, door te proberen opzettelijk verkeerde feiten in te voeren, en laat de reactie van het systeem zien. Voeg de resultaten toe aan het verslag (onder Resultaten).
5. Schrijf query’s voor de 3 informatiebehoeften uit de casus.
6. Bedenk 3 aanvullende aanvragen die de sterke punten van het één of het andere systeem onderstrepen. Neem zowel de code als de resultaten van deze query's op in je verslag.
7. Maak een tweede implementatie in de Neo4j omgeving met dezelfde voorbeeldgegevens (zie stap 2).
8. Zorg ervoor dat alle integriteitsregels (inclusief mandatory child references) ook bij de tweede implementatie gehandhaafd worden (zie stap 3).
9. Toon aan dat de Neo4j implementatie alle integriteitsregels handhaaft, door opzettelijk verkeerde feiten in te voeren, en laat de reactie van het systeem zien. Voeg deze toe aan het verslag (onder Resultaten).
10. Schrijf query's voor de Neo4J-database, om daarmee aan dezelfde informatiebehoeften te voldoen (stap 5 en 6), en controleer of deze dezelfde resultaten opleveren. Neem ook de resultaten van deze query's op in het verslag.

Taak 5: Vergelijkend onderzoek

Het doel van deze onderzoeksopdracht is om positieve en negatieve aspecten van de twee implementaties te ontdekken en deze te vergelijken. Formuleer deelvragen waarmee je de hoofdvraag kunt beantwoorden. Kies hiervoor de aspecten waarop je de implementaties wilt vergelijken (b.v.: de gelijkwaardigheid van de implementaties, de manier om business rules te implementeren, de complexiteit van query's om aan een informatiebehoefte te voldoen, opslag, performance, ...) en wat jullie criteria zijn.

Welke technologie heeft jouw/jullie voorkeur en waarom?

Taak 6: Onderzoeksverslag

Schrijf een onderzoeksverslag over taak 3 en 4 (ervaringen met en vergelijking van de implementaties) met:

- **Abstract**: een korte samenvatting van het verslag.
- **Inleiding**, met daarin een de voorgegeven hoofdvraag en je eigen deelvragen.
- **Methoden**, waarin je beschrijft hoe je het onderzoek hebt aangepakt (hier nog GEEN resultaten!). De aanpak moet aansluiten bij je deelvragen.
    - Hoe heb je de structuur, de inhoud (populatie) en de gelijkwaardigheid van de twee databases vastgelegd?
    - Hoe ben je bij de informatievragen en de query's uitgekomen?
    - Hoe heb je de vergelijking aangepakt (aspecten en criteria)?
- **Theoretisch kader:** je resultaten van je literatuuronderzoek.
- **Resultaten**, hierin vermeld je wat je hebt gevonden:
    - de beschrijving van de structuur en inhoud van de databases (en bewijzen dat constraints gehandhaafd worden)
    - de resultaten van de query's
    - de uitkomst van de vergelijking
- **Discussie**, waarin je de resultaten bespreekt (met welke aannames heb je gewerkt, zijn de onderzoeksvragen voldoende beantwoord, etc.) en argumenten geeft voor je conclusies.
- **Conclusies**, waarin je kort de antwoorden op de onderzoeksvragen en eventuele andere ontdekkingen weergeeft. Let op: de argumenten voor alle conclusies staan onder Discussie; hier vermeld je alleen de conclusies zelf.
- **Referentielijst** in APA-formaat.

Feedback

Je docent is beschikbaar voor feedback gedurende de geroosterde lesuren. Het advies is om vooral feedback te vragen rondom de te formuleren onderzoeksvragen.