Bericht

Erstelle einen Bericht in dem Du das Resultat reflektierst. Darin sollen folgende Fragen beantwortet werden. Ein Worddokument mit dem Bericht.

### Was wird erwartet

Das konzeptionelle Datenmodell. Dieses darf von Hand gezeichnet sein oder mit einem Tool erstellt werden. Weiter müssen die Skripte zur Erstellung der beiden Datenbanken (SQL und NoSQL) als Text oder .sql Datei abgegeben werden.

### Bewertungskriterien für die Benotung der Abgabe

Das Beispiel umfasst 2 bis 10 Seiten (ohne Datenmodell). Er beantwortet alle Fragen schlüssig, fundiert und verständlich.

### Aufgabenstellung zum Thema Vergleich von Datenbanken

Ziel: Anhand einer konkreten Problemstellung wird eine Lösung mit NoSQL und mit SQL verglichen. Überlege Dir eine Problemstellung, die sinnvollerweise mit einer NoSQL Datenbank umgesetzt wird. Die Problemstellung darf nicht trivial sein, soll also in SQL mindestens einen Fremdschlüssel enthalten. Sei es zwischen zwei Tabellen oder als Selbstreferenz. Wähle eine NoSQL Datenbank aus, die zur Problemstellung passt.

Halte fest, welche Vorteile die gewählte Datenbank hat und wie Du diese Vorteile messen kannst. Entwerfe das konzeptionelle Datenmodell und setze dieses nachher mit der gewählten NoSQL Datenbank und mit einer relationalen Datenbank um. Die Umsetzung soll so gemacht werden, dass diese auf die Problemstellung optimiert ist. Das bedeutet, dass das logische und physische Datenmodell den Abfragebedürfnissen angepasst ist. Fülle beide Datenbanken mit denselben Daten. Vergleiche die beiden Datenbanken mit den definierten Messungen.

# Spezifische Fragen

### Welche NoSQL Datenbank hast Du gewählt und wieso diese?

MongoDB ist die bekannteste der NoSQL-Datenbanken. Da wir ihr in der Zukunft wohl noch öfters begegnen werden, machte es am meisten Sinn MongoDB etwas genauer anzuschauen. Dokument-Datenbanken scheinen das breiteste Anwendungsgebiet der NoSQL-Datenbanken zu haben. Die flexible Struktur der Dokumente ist ein grosser Vorteil gegenüber SQL.

### Wie sieht die Problemstellung aus?

### Wie sieht das konzeptionelle Datenmodell aus?

ERD ERM

### Wie sieht die Umsetzung in den beiden Datenbanken aus?

{

"\_id": { "$oid": "61d4a8b47e51883717060e1a" },

"post\_id": { "$numberInt": "4" },

"title": "Here buy TV at anyone.",

"description": "East despite work perform first.",

"url": "http://www.clark.com/",

"likes": { "$numberInt": "40495" },

"author": "williamsjennifer",

"email": "rodgersheather@yahoo.com",

"date": { "$date": { "$numberLong": "1614290590000" } },

"tags": ["#Here", "#buy", "#TV", "#at", "#anyone"],

"comments": [

{

"comment\_id": { "$numberInt": "37" },

"comment\_author": "timothymayer",

"comment\_author\_email": "isaiah02@hotmail.com",

"comment\_date": { "$date": { "$numberLong": "1635125256000" } },

"comment\_text": "Into.",

"comment\_likes": { "$numberInt": "11" }

},

{

"comment\_id": { "$numberInt": "38" },

"comment\_author": "fowlertodd",

"comment\_author\_email": "gholt@hotmail.com",

"comment\_date": { "$date": { "$numberLong": "1605428573000" } },

"comment\_text": "Be.",

"comment\_likes": { "$numberInt": "772" }

}

]

}

### Welche Vorteile hast Du bei der NoSQL Datenbank erwartet?

Auch das Einrichten eines Clusters und die Verbindung mit Python hat sich als Einfach herausgestellt.

Der Wichtigste Vorteil gegenüber SQL ist jedoch die Dynamische Entwicklung und die hohe Skalierbarkeit der Datenbank von einzelnen Servern bis hin zu komplexen Architekturen über mehrere Rechenzentren. Da die Dokumente sehr unterschiedlich sein können, ist man in der Weiterentwicklung der Datenbank extrem flexibel. In Unserem Testdatensatz zu Filmen gibt es für die älteren Filme deutlich weniger Informationen als zu den Neueren. Dies ist für MongoDB jedoch kein Problem.

Bei den Abfragen sind keine Joins nötig was die Abfragen theoretisch weniger Komplex macht. Die grösse eines Objekts oder Arrays innerhalb eines Dokumentes ist nicht vorgegeben und kann bei jedem Dokument unterschiedlich sein. Sogar die Struktur des Dokuments selbst kann stark unterschiedlich sein. MongoDb ist also extrem flexibel. Generell ist der Entwicklungsaufwand kleiner als bei SQL. Die Datenbank kann relativ leicht erweitert und verändert werden, weil sie schemalos ist. Auch ist die Struktur, da in JSON (respektive BSON) Frontend tauglich und erleichtert somit die Implementation.

Dabei sehe ich die Vorteile von MongoDB vor allem bei der Abfragezeit, weil zum Darstellen auf eines Blog-Posts genau ein gesamtes Dokument abgerufen werden muss. Zudem sind Abfragen so auch sehr simpel. Weitere Vorteile sind die Flexibilität und Einfachheit, die Datenstruktur anzupassen und ganz allgemein die Implementationszeit.

### Welche Messkriterien hast Du festgelegt?

Implementationszeit (Welche Vorkenntnisse habe ich mit den beiden Datenbanken)

Lines of Code

Abfragezeit Dokument

Abfragezeit vom Metadaten des Posts für häufigster Tag

Einfachheit die Architektur zu verändern

Speicherbedarf

### Wie sehen die Messresultate aus?

### Interpretiere und reflektiere diese Messresultate.

# Fazit

### Was sind Deine Vorkenntnisse in Datenbanken aus Deiner beruflichen Erfahrung?

Ich habe keine berufliche Erfahrung mit Datenbanken. Jedoch habe ich in dem Data Science Studium schon ein paar SQL-Datenbanken genutzt. Ich habe ebenfalls in der Form des Modul GDB schon mit Datenbanken Erfahrungen sammeln können. Jedoch habe ich vor dieser Abgabe noch keine eigenen Datenbanken aufgesetzt und eigene Daten eingelesen.

### Gibt es andere wichtige Punkte, die auf Deinen Lernerfolg einen wichtigen Einfluss hatten?

### Was war für Dich die wichtigste Erkenntnis? (Des gesamten Projekts)