

Mongomir Pytkovic

Migration einer SQLite-Datenbank nach MongoDB mit Python

Gruppe 7

L. Arnold, P. Bucher, C. J. Christensen, J. Kaiser, M. Werthmüller

Modul DBS

Herbstsemester 2017

26.11.2017

Inhaltsverzeichnis

1 Einführung	2
1.1 Was ist der Kontext, warum ist das Projekt relevant, und worum geht es? . . .	2
2 Datenmanagement	2
2.1 Um welche Datenbanktechnologie handelt es sich?	2
2.2 Welche Anwendungen (Use Case) unterstützt ihre Datenbank?	2
2.3 Welche Daten werden migriert/eingefügt, und wie genau?	2
2.4 Wie interagiert der Benutzer mit der Datenbank?	3
3 Datenmodellierung	3
3.1 Welches Datenmodell (ER) liegt ihrem Projekt zugrunde?	3
3.2 Wie wird ihr Datenmodell in Ihrer Datenbank in ein Schema übersetzt?	4
4 Datenbanksprachen	4
4.1 Wie werden Daten anhand einer Query abgefragt?	4
5 Konsistenzsicherung	4
5.1 Wie wird die Datensicherheit gewährleistet?	4
5.2 Wie können Transaktionen parallel/konkurrierend verarbeitet werden?	4
6 Systemarchitektur	4
6.1 Wie ist der Server aufgebaut und wie wurde er installiert?	4
6.2 Wie kann die Effizienz von Datenanfragen optimiert werden?	4
7 Vergleich mit relationalen Datenbanken	4
7.1 Vergleichen Sie ihre NoSQL-Technologie mit SQL-Datenbanken.	4

8 Schlussfolgerungen	4
8.1 Was haben Sie erreicht, und welche Erkenntnisse haben sie dabei gewonnen?	4
8.2 Wie beurteilt ihre Gruppe die gewählte Datenbanktechnologie, und was sind Vor- und Nachteile?	4

1 Einführung

1.1 Was ist der Kontext, warum ist das Projekt relevant, und worum geht es?

Datensätze von Fussballspielen von Kaggle (SQLite), Migration eines Teils davon nach MongoDB mit Python

2 Datenmanagement

2.1 Um welche Datenbanktechnologie handelt es sich?

SQLite -> MongoDB

2.2 Welche Anwendungen (Use Case) unterstützt ihre Datenbank?

- Abfragen sämtlicher Spiele, in denen ein bestimmter Spieler zum Einsatz gekommen ist

2.3 Welche Daten werden migriert/eingefügt, und wie genau?

Es sollen die relationalen Entitäten Match, Player, Player_Attributes, Team, Team_Attributes zu einer JSON-Struktur namens Match kombiniert werden. Dabei werden die an einem Spiel teilnehmenden Mannschaften mit ihren Spielern nicht referenziert, sondern direkt mit allen relevanten Informationen in einer hierarchischen Struktur abgelegt:

- match
 - home_team
 - * name
 - * players
 - player_1
 - ...
 - player_n
 - away_team
 - * name
 - * players

- player_1
- ...
- player_n
- goals_home_team
- goals_away_team

Die Entitäten League und Country sollten zudem in eine League-Struktur überführt werden:

- league
 - id
 - name
 - country

2.4 Wie interagiert der Benutzer mit der Datenbank?

TODO: per Web-Schnittstelle

3 Datenmodellierung

3.1 Welches Datenmodell (ER) liegt ihrem Projekt zugrunde?

Dies ist ein Auszug aus dem ER-Modell, der nur die Tabellen und Spalten enthält, die auch tatsächlich in die Dokumentdatenbank migriert werden sollen:

- League
 - TODO
- Match
- Player
- Player_Attributes
- Team
- Team_Attributes

Das komplette Schema ist auf [Kaggle](#) ersichtlich.

3.2 Wie wird ihr Datenmodell in Ihrer Datenbank in ein Schema übersetzt?

4 Datenbanksprachen

4.1 Wie werden Daten anhand einer Query abgefragt?

5 Konsistenzsicherung

5.1 Wie wird die Datensicherheit gewährleistet?

Gar nicht; die Daten sind nicht sensibel. Evtl. Userkonfiguration, sodass nur ein Benutzer die Migration vornehmen kann, der Enduser aber nur abfragen kann.

5.2 Wie können Transaktionen parallel/konkurrierend verarbeitet werden?

Es sind nur lesende Abfragen möglich.

6 Systemarchitektur

6.1 Wie ist der Server aufgebaut und wie wurde er installiert?

Siehe Dockerfile

6.2 Wie kann die Effizienz von Datenanfragen optimiert werden?

7 Vergleich mit relationalen Datenbanken

7.1 Vergleichen Sie ihre NoSQL-Technologie mit SQL-Datenbanken.

8 Schlussfolgerungen

8.1 Was haben Sie erreicht, und welche Erkenntnisse haben sie dabei gewonnen?

8.2 Wie beurteilt ihre Gruppe die gewählte Datenbanktechnologie, und was sind Vor- und Nachteile?