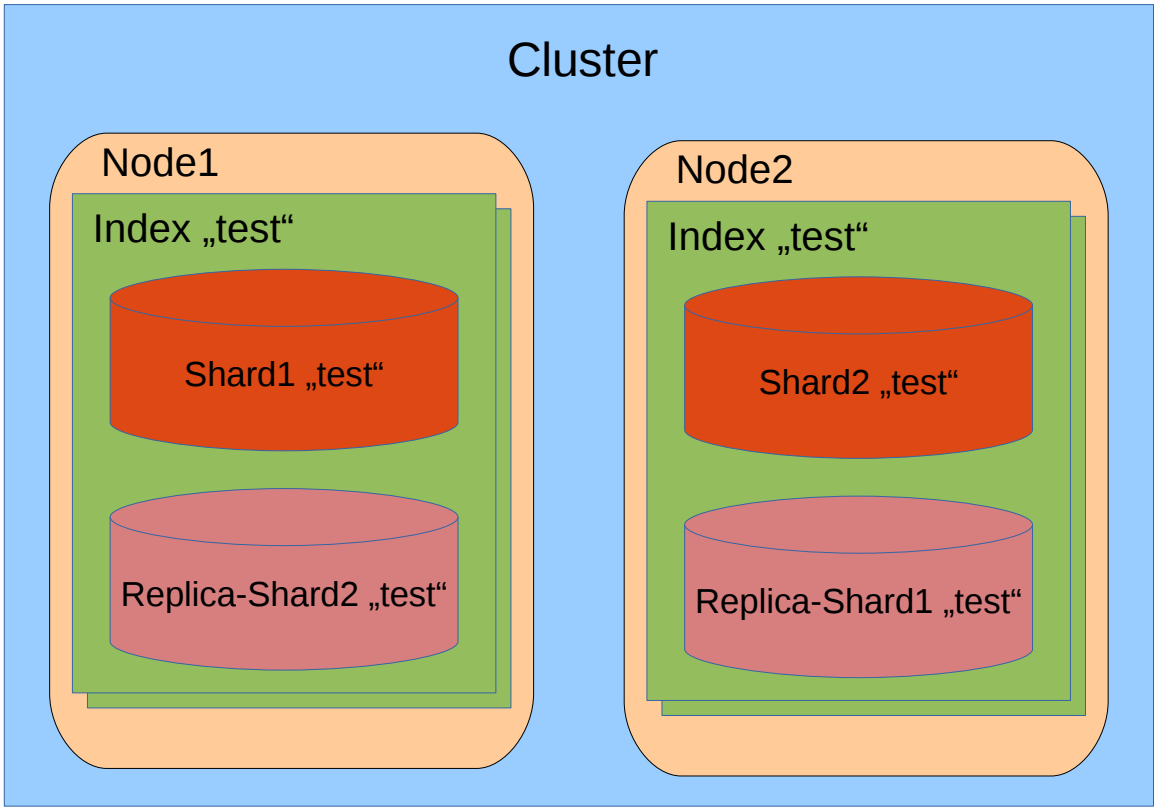
Elasticsearch 101

Agenda

- ES Architecture
- CRUD
- Basic Queries
- Similarity Search
- Full-Text Search

Basic Concepts

- Near Realtime
- Cluster
- Node
- Index
- Document
- Shards



Task 1 - CRUD

- Create a new index called „cars“
  - Make sure your cluster is green after this request
  - Set number_of_shards to 2
- Insert a document like this with ID=1:
  - {"brand":"BMW","series":"5","color":"GREEN","power_kw":100,"marketing_text":"Bringing together BMW’s heritage of innovation and design excellence to perform stunningly, the BMW 5 Series grabs attention for all the right reasons."}
- Find your inserted car by ID

Task 2 – Basic Queries

- Add some more documents to the cars-index. You can do this with one request ;)
  - {"brand":"BMW","series":"3","color":"BLUE","power_kw":150,"marketing_text":"Since 1975, the BMW 3 Series has led the field. With unwavering dedication to the journey, it brings together advanced engineering and expert craftsmanship to create The Ultimate Driving Machine."}
  - {"brand":"VW","series":"Golf","color":"BLACK","power_kw":90,"marketing_text":"The iconic Golf now with pioneering new engines, keener design, new assistance systems and a completely new generation of infotainment systems."}
  - {"brand":"Toyota","series":"Prius","color":"BLACK","power_kw":120,"marketing_text":"Be in your element."}
  - {"brand":"Ford","series":"Mustang","color":"RED","power_kw":250,"marketing_text":"Mustang is designed to custom-fit the way you drive, down to the last detail. From the look of the instrument panel to the sound of its growl to the way it feels tackling a curve, this legend was born to make your own. Plus, Mustang is the 2018 Highest Ranked Midsize Sporty Car in Initial Quality."}
  - {"brand":"Mercedes-Benz","series":"E-Class","color":"GRAY","power_kw":160,"marketing_text":"Mercedes-Benz takes a big step into the future with the new E-Class. The tenth-generation business saloon delivers stylish highlights with its distinct, emotive design and high-grade interior. The new E-Class also marks the world premiere of numerous technical innovations. They enable comfortable, safe driving on a new level plus a new dimension in driver assistance – among other things. A generation of completely new four-, six-, and eight-cylinder engines is just one of many highlights of the new E-Class. The sum total of its innovations make the E-Class the most intelligent saloon in the business class."}
  - {"brand":"Audi","series":"R8","color":"RED","power_kw":419,"marketing_text":"Thrilling performance. Breathtaking handling. The new Audi R8 Coupé has arrived."}
  - {"brand":"Opel","series":"Insignia","color":"BLUE","power_kw":147,"marketing_text":"Stylish design. Superb connectivity. Smart driver-assistance. Insignia Grand Sport is your premium invitation to the open road."}
- Count the number of documents in your index
- Find all BLACK cars
- Find all cars with more than 100kw power and order by power_kw desc

Indices / Mapping / Datatypes

- ES tries to identify the datatype and all fields are indexed by default
- existing field mappings can‘t be updated!
  - ignore_above can be changed
  - New fields can be added
  - Additional „multi-field“ mappings can be added to existing fields (add additional index to existing field)

Task 3 – Custom Mapping

- Inspect the auto-created cars index mapping
- Delete the cars index
- Create a custom cars index fitting to our data/requirements
- Insert the data from the previous tasks
- Retry the queries from Task 2

Similarity Seach

- Elasticsearch provides a _***score*** property for each match in a query result.
- In the mapping you can define a ***boost*** for each datatype-index.
- In a search request you can set a ***boost*** for a query, which overrides the defined boost in the mapping.
- Similarity Search Query types:
  - More Like This Query
  - Bool Query (should)

Task 4 – Similarity Search

- Build a Bool-Query to find similar cars for a search request:
  - power_kw should be greater than 100, this is most important.
  - color should be BLACK or BLUE, this is mid important.
  - brand should be BMW, this is less important.
- Build a More Like This Query to find similar cars...
  - Search in "marketing_text", "brand" and "series" for “BMW”
    - HINT: set *min_term_freq* and *min_doc_freq* to 1
  - for a existing document in the cars index (_id=1 from Task 1) for “brand”, “color” and “series”
    - HINT: set *min_term_freq* and *min_doc_freq* to 1

How does ES indexing work?

- Es stores data in a inverted index
  https://www.elastic.co/guide/en/elasticsearch/guide/current/inverted-index.html
- Text is analyzed
  https://www.elastic.co/guide/en/elasticsearch/reference/current/analyzer-anatomy.html
  - 1. Character filters
  - 2. Tokenizer
  - 3. Token filters

Task 5 – Full-Text Search

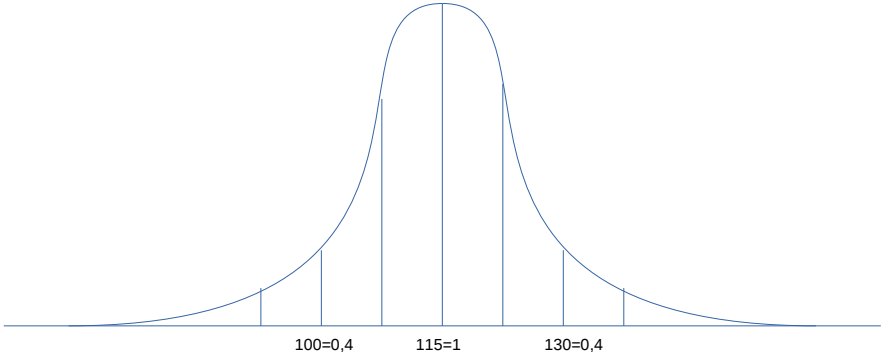
- Add an additional type mapping for field “marketing_text” and use the “english” analyzer.
- Search for “innovative car” in “marketing_text” and use the previously created type mapping.
- Also compare the results with the default analyzer used in the default mapping for “marketing_text”.

Bonus Task 1 – Aggregations

- Create a Terms Aggregation for color to get all available colors with the corresponding count of documents.
- Create a Aggregation over “power_kw” to display min/max/average.

Bonus Task 2 – Function Score

- You want to search for power_kw = 115, but you want also to score cars which are near by the provided value, but with a lower score. Build a Query for that requirement.



Bonus Task 3 – add a new Field

- You want to add a new Field “power_ps” which is calculable from power_kw.
  - First update the mapping.
  - Then create a “migrate script” which calculates the new field value from power_kw and re-indexes the documents. HINT: use a painless script.

Bonus Task 4 – Source filtering

- Search for all cars and exclude “marketing_text” from the response.
- Search for all cars and include only “brand” and “series” in the response.