UniRitter
Laureate International Universities®

Banco de Dados NoSQL

(NS0301)

Prof. Giovane Barcelos
giovane_barcelos@uniritter.edu.br

O que vamos aprender?

- **Importância e características chaves do ElasticSearch**
- **Instalação**
- ***CRUD* básico**
- **Pesquisas (*queries*)**
- ***ELK***

Quem é o ElasticSearch? (https://www.elastic.co/)

- Banco de dados *NoSQL* de Pesquisa (*Search Engine*) orientado a Documento
- *Elasticsearch* é um servidor de buscas distribuído e uma *API* de indexação invertida de documentos baseado no *Apache Lucene* e desenvolvido em *Java*
- Foi desenvolvido por Shay Banon e disponibilizado sobre os termos Apache da *Licença Apache*

Porque utilizar o ElasticSearch?

- **Fácil de Escalar (Distribuído)**
- **Tudo e todas as chamadas são em *JSON* (API RESTful)**
- **Poder do Lucene sob o capô**
- **Excelente *Query DSL* (*Domain-Specific Language*)**
- ***Multi-tenancy* (Multi-inquilino)**
- **Suporte para recursos de pesquisa avançada (*Full Text*) por Lista Invertida**
- **Configurável e Extensível**
- **Orientado a Documento**
- **Sem Esquema**
- **Possui gerenciamento de conflitos**
- **Comunidade muito ativa**

Porque *ElasticSearch* usa Lista Invertida?

- Porque uma lista invertida é uma lista (ou índice) ordenada de chaves, onde cada chave contém uma ligação para os documentos que a contém, ou seja, uma chave associada as suas frequências
- Supondo a seguinte lista de documentos:

  1: "Estudar ElasticSearch é divertido"

  2: "Divertido é *NoSQL*"

  3: "Divertido mesmo é não estudar"
- Lista Invertida obtida dos documentos:

estudar: {1, 3}	elasticsearch: {1}
é: {1,2,3}	divertido: {1,2,3}
nosql: {2}	mesmo: {3}
não: {3}	

Quais os conceitos chaves do ElasticSearch? {1-2}/10

1. *Cluster*:
   - ✔ Consiste em um ou mais nós que compartilham o mesmo nome de cluster
   - ✔ Cada cluster tem um único nó mestre que é escolhido automaticamente pelo cluster e que pode ser substituído se o nó mestre atual falhar
2. Node:
   - ✔ Um nó é uma instância em execução do ElasticSearch que pertence a um cluster
   - ✔ Vários nós podem ser iniciados em um único servidor para fins de teste, mas geralmente você deve ter um nó por servidor
   - ✔ Na inicialização, um nó usará unicast (ou multicast, se especificado) para descobrir um cluster existente com o mesmo nome de cluster e tentará se juntar a esse cluster

Quais os conceitos chaves do ElasticSearch? {3-4}/10

3. *Index*:
   - ✔ Um índice é como um "banco de dados" em um banco de dados relacional. Tem um mapeamento que define vários tipos
   - ✔ Um índice é um espaço de nomes lógico que mapeia para um ou mais shards (fragmentos) primários e pode ter zero ou mais fragmentos de réplica
4. *Type*:
   - ✔ Um tipo é como uma "tabela" em um banco de dados relacional
   - ✔ Cada tipo tem uma lista de campos que podem ser especificados para documentos daquele tipo
   - ✔ O mapeamento define como cada campo no documento é analisado

Quais os conceitos chaves do ElasticSearch? 5/10

5. ***Document*:**

- ✔ **Um documento é um documento *JSON* que é armazenado no *ElasticSearch***
- ✔ **É como uma linha em uma tabela em um banco de dados relacional**
- ✔ **Cada documento é armazenado em um índice e tem um tipo e um id**
- ✔ **Um documento é um objeto *JSON* que contém zero ou mais campos ou pares de chave-valor**
- ✔ **O documento *JSON* original indexado será armazenado no campo _*source*, que é retornado por padrão ao obter ou procurar por um documento.**

Quais os conceitos chaves do ElasticSearch? 6/10

6. *Field*:

- ✔ **Um documento contém uma lista de campos ou pares de valores-chave**
- ✔ **O valor pode ser um valor simples (string, inteiro, data, ...), ou uma estrutura aninhada como uma matriz ou um objeto**
- ✔ **Um campo é semelhante a uma coluna em uma tabela em um banco de dados relacional**
- ✔ **O mapeamento para cada campo tem um campo "tipo" (não confundir com tipo de documento) que indica o tipo de dado que pode ser armazenado nesse campo, por exemplo, número inteiro, string e objeto**
- ✔ **O mapeamento também permite que você defina, entre outras coisas, como o valor de um campo deve ser analisado**

Quais os conceitos chaves do ElasticSearch? 7/10

7. *Mapping*:

- ✔ **Um mapeamento é como uma "definição de esquema" em um banco de dados relacional**
- ✔ **Cada índice tem um mapeamento, que define cada tipo dentro do índice, além de um número de configurações de todo o índice**
- ✔ **Um mapeamento pode ser definido explicitamente ou será gerado automaticamente quando um documento é indexado**

Quais os conceitos chaves do ElasticSearch? 8/10

8. *Shard*:

- ✔ **Um fragmento (*shard)* é uma única instância do *ElasticSearch***
- ✔ **É uma unidade de "trabalho" de baixo nível que é gerenciada automaticamente pelo *ElasticSearch***
- ✔ **Um índice é *namespace* lógico que aponta para *shards* primários e de réplica**
- ✔ **O Elasticsearch distribui *shards* entre todos os nós no *cluster* e pode mover *shards* automaticamente de um nó para outro no caso de falha do nó ou na adição de novos nós**

Quais os conceitos chaves do ElasticSearch? 9/10

9. *Primary Shard*:

- ✔ **Cada documento é armazenado em um único fragmento primário**
- ✔ **Quando você indexa um documento, ele é indexado primeiro no *shard* primário, em seguida, em todas as réplicas do fragmento primário**
- ✔ **Por padrão, um índice tem 5 fragmentos primários**
- ✔ **Você pode especificar menos ou mais *shards* primários para dimensionar o número de documentos que seu índice pode manipular**

Quais os conceitos chaves do ElasticSearch? 10/10

10. ***Replica Shard*:**

- ✔ **Cada fragmento primário pode ter zero ou mais réplicas**
- ✔ **Uma réplica é uma cópia do fragmento primário e tem dois propósitos:**
  - ***Failover*: um fragmento de réplica pode ser promovido a um fragmento primário se o primário falhar**
  - **Desempenho: obter e solicitar pesquisas que podem ser manipuladas por *shards* primários ou de réplica**

ElasticSearch vs Banco Relacional

ElasticSearch	Banco Relacional
Índice (*Index*)	Esquema / Banco de Dados
Type	Tabela
Documento (*JSON*)	Linha
Campo (*Field*)	Coluna
Mapeamento (*Mapping*)	Estrutura da Tabela
Query DSL	SQL

Como instalar o *ElasticSearch*?

- Fazer download do ES (*ElasticSearch*): https://www.elastic.co/downloads/elasticsearch
- Descompactar e executar: bin/elasticsearch
- Como serviço no Linux executar:

  apt-get install elasticsearch # Debian / Ubuntu

  yum install elasticsearch # Centos e assemelhados
- Para verificar se esta rodando acessar a URL http://localhost:9200/
- Baixar e Descompactar o Kibana
- Executar o Kibana: bin/kibana

- Ferramentas Auxiliares:

  Instalar o Plugin Sense do Mozilla Firefox

  Instalar o Plugin Dejavu do Chromium

O que foi instalado ao descompactar?

1. bin/ - scripts binários para iniciar e parar cada nó
2. config/ - arquivos de configuração, tais como, o elasticsearch.yml e o logging.yml, bem como, configurações de tamanho de *heap* e quantidade de descritores de arquivos
3. data/ - onde se localiza os arquivos de dados para cada índice, *shard* alocado em cada nó
4. lib/ - bibliotecas em Java do ES que podem ser integradas com outros aplicativos
5. logs/ - registros de *log* de execução, indexação e pesquisa
6. modules/ - bibliotecas e configurações dos módulos
7. plugins/ - plugins externos instalados no banco de dados

Porque utilizar o ElasticSearch?

- **Fácil de Escalar (Distribuído)**
- **Tudo e todas as chamadas são em *JSON* (API RESTful)**
- **Poder do Lucene sob o capô**
- **Excelente *Query DSL* (*Domain-Specific Language*)**
- ***Multi-tenancy* (Multi-inquilino)**
- **Suporte para recursos de pesquisa avançada (*Full Text*) por Lista Invertida**
- **Configurável e Extensível**
- **Orientado a Documento**
- **Sem Esquema**
- **Possui gerenciamento de conflitos**
- **Comunidade muito ativa**

Como interagir com o *ElasticSearch*?

- ***API RESTful***
  - **Porta padrão 9200**
- **API Java**
  - **Cliente "nó": cria uma instância do *elasticsearch* e se associa ao *cluster*, porém não armazena dados**
  - **Cliente "transporte": comunica-se remotamente com algum nó do *cluster***

Indexando documentos no ES

- Utiliza-se a API de Indexação
  - ✔ Id auto-gerado: POST /{indice}/{tipo}
  - ✔ Id externo: PUT /{indice}/{tipo}/{id}
- Documentos são imutáveis – para alterá-los é necessário substituir todo o conteúdo
  - ✔ Endpoint + Id interno: PUT /{indice}/{tipo}/{id}
  - ✔ Endpoint _update existe para facilitar o processo: POST /{indice}/{tipo}/{id}/_update

Indexando documentos no ES

- Utiliza-se a API de Indexação
  - ✔ Id auto-gerado: POST /{indice}/{tipo}
  - ✔ Id externo: PUT /{indice}/{tipo}/{id}
- Documentos são imutáveis – para alterá-los é necessário substituir todo o conteúdo
  - ✔ Endpoint + Id interno: PUT /{indice}/{tipo}/{id}
  - ✔ Endpoint _update existe para facilitar o processo: POST /{indice}/{tipo}/{id}/_update
- Como saber se o documento já existe?

  HEAD /{indice}/{tipo}/{id}
- Como saber se um documento será criado?

  PUT /{indice}/{tipo}/{id}/_create
- Como remover um documento?

  DELETE /{indice}/{tipo}/{id}

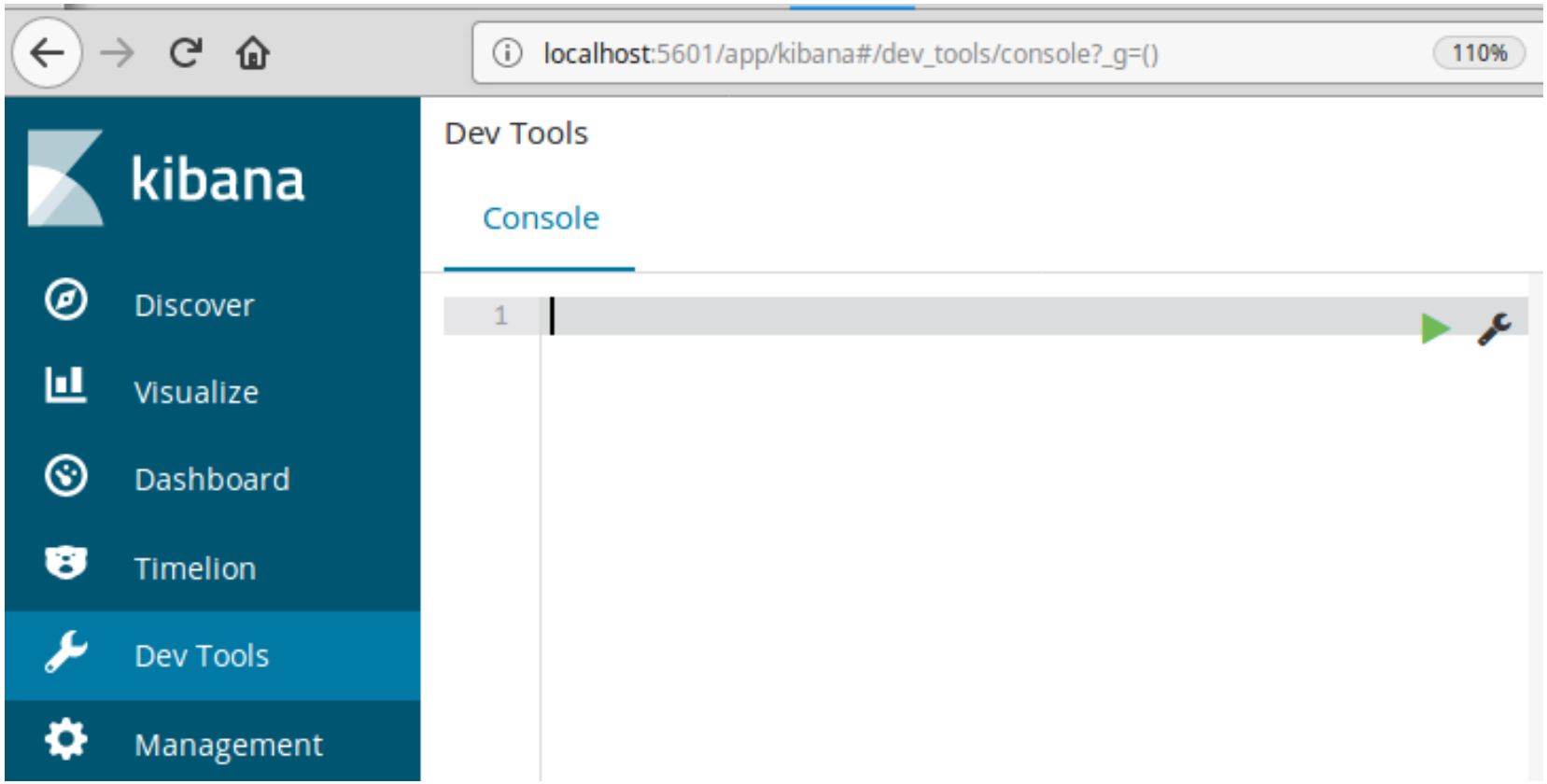
Indexando documentos no ES

- Como garantir a consistência das atualizações?
  - Elasticsearch cria um campo de versão para cada documento ("_version")
  - A aplicação pode passar como parâmetro a versão para aplicar as atualizações de dados

    PUT /{indice}/{tipo}/{id}?version=1
  - Caso a versão tenha sido alterada, a requisição falha com status 409 Conflict

Vamos usar o Kibana

- **No navegador: http://localhost:5601**
- **Vá para o menu "Dev Tools"**



Comandos do ElasticSearch

- **Como verificar todos os Índices?**

  GET _cat/indices

- **Como criar um Índice?**

  PUT frasesfamosas

- **Como Inserir/Indexar Documentos?**

  POST /frasesfamosas/frase/1

```
{
  "autor": "Benjamin Franklin",
  "frase": "A tragédia da vida é que ficamos velhos cedo demais. E sábios, tarde demais."
}
```

Comandos do ElasticSearch

- **Vamos Inserir/Indexar mais um documento**

```
POST /frasesfamosas/frase/2
{
   "autor": "Steve Jobs",
   "frase": "Lembrar que você vai morrer é a melhor maneira que eu conheço para evitar a armadilha de pensar que você tem algo a perder. Você está nu. Não há razão para não seguir seu coração."
}
```

Comandos do ElasticSearch

- **Vamos Inserir/Indexar mais outro documento**

  POST /frasesfamosas/frase/3

  {

  "autor": "Albert Einstein",

  "frase": "Lembre-se que as pessoas podem tirar tudo de você, menos o seu conhecimento."

  }

- **Como verificar todos os documentos inseridos e suas propriedades?**

  GET /frasesfamosas/frase/_search

Comandos do ElasticSearch

- **Como Remove um Documento?**

  DELETE /frasesfamosas/frase/1

- **Como alterar um Documento?**

```
POST /frasesfamosas/frase/3/_update
{
  "doc": {
    "frase": "Cada dia sabemos mais e entendemos menos."
  }
}
```

Como funcionam as pesquisas?

- Funcionalidade principal da ferramenta
- Busca estruturada
  - Busca por algum campo específico ordenando resultados por algum outro campo
  - Similar a uma query SQL
- Busca textual
  - Encontrar documentos que contenham certas palavras chaves, ordenando por relevância
- Busca mista
  - Combinação entre os dois anteriores

Quais as formas de pesquisa?

- **Busca via URI**
  - **Geralmente usada para testes ou buscas simples**
  - **Opções limitadas**
  - **GET /{indice}/{tipo}/_search?q={query}**
- **Busca via Request Body**
  - **Utiliza Query DSL**
    - **Linguagem de query específica do Elasticsearch**
    - **Define uma série de queries e filtros e permite a combinação dos mesmos**
  - **POST|GET /{indice}/{tipo}/_search**

Quais as formas de pesquisa?

- **Filtro**
  - **Usado para buscar campos que contém valores exatos**
  - **Buscas que podem ser respondidas com sim ou não**
  - **Exemplos: terms, range, exists, bool, etc**
- **Query**
  - **Usado para buscas textuais**
  - **Buscas cujo resultado dependam de relevância**
  - **Exemplos: match, bool, etc**

Quais as formas de pesquisa?

- **Analisadores**
  - **Quando um documento é indexado, os campos texto são "quebrados" em termos que podem ser buscados**
  - **A "quebra" é feita por um analisador**
    - **É possível criar novos ou usar algum pré-definido**
    - **Analisador padrão é aplicado a todos os campos textos caso nenhum seja definido**
      - **A associação é feita no mapeamento dos tipos**
  - **Posteriormente, a busca feita no campo deve passar pelo mesmo analisador para garantir a consistência dos resultados**
- **Paginação**
  - **Padrão de 10 resultados por página**

Quais as formas de pesquisa?

- **Ordenação**
  - **Feita por algum campo específico ou por relevância**
- **Relevância**
  - **Cada documento voltará com um campo especial chamado "_score" que contém um decimal representando a relevância**
  - **Quanto maior o número, maior a relevância**
  - **Relevância é calculada de acordo com um algoritmo de similaridade**
  - **O algoritmo padrão utilizado é o Term Frequency/Inverse Document Frequency (TF/IDF)**

Comandos do ElasticSearch de Pesquisa

- **Quais são todos os índices?**

  http://localhost:9200/_search
- **Quais todos os tipos do índice frasesfamosas?**

  http://localhost:9200/frasesfamosas/_search
- **Quais os documentos do tipo frase?**

  http://localhost:9200/frasesfamosas/frase/_search
- **Mostrar os documentos do tipo frase paginado?**

  http://localhost:9200/frasesfamosas/frase/_search?size=1&from=2
- **Quais são os documentos que possuem a string "pessoas"?**

  GET frasesfamosas/frase/_search

  {

  "query": { "query_string": { "query": "pessoas" }}}

Comandos do ElasticSearch de Pesquisa

- Quais os documentos que tem o autor "Albert Einstein"?

```
GET /frasesfamosas/frase/_search
{ "query": {
 "match": {
"autor": "Albert Einstein"
}}}
```

- Quais os documentos que tem o autor ou no própria frase a string "Albert Einstein"?

```
GET /frasesfamosas/frase/_search
{ "query": {
 "multi_match": {
"query": "Albert Einstein",
"fields": ["frase","autor"]}}}
```

O que vamos ver no Próximo Encontro

- O Elasticsearch pode ser utilizado em conjunto com outras duas ferramentas, que juntas formam o stack ELK
  - Elasticsearch
  - Logstash
  - Kibana

Lembre-se

" No fim tudo dá certo, e se não deu certo é porque ainda não chegou ao fim. "

Fernando Sabino