



TECHNICAL UNIVERSITY OF LIBEREC  
Faculty of Mechatronics, Informatics  
and Interdisciplinary Studies



# DATABÁZE PRO PROHLEDÁVÁNÍ A ANALÝZU TEXTU

*Lukáš Matějů*

16.4.2024 | DPB



# ČÁST I.: OPAKOVÁNÍ



# OPAKOVÁNÍ

- **agregace**
  - dávkové zpracování dokumentů vracející kompaktní výsledek i po provedení celé řady operací
    - první fáze – shluknutí hodnot z různých dokumentů
    - druhá fáze – provedení operací na shluknuty datech a vrácení výsledku
  - tři možnosti v MongoDB
    - agregační roura (pipeline)
      - dokumenty jsou zpracovávány postupně v krocích až do konečného výsledku
    - map-reduce (dnes již deprecated)
      - map – mapování
        - zpracování dokumentů do objektů odpovídajících vstupním dokumentům
      - reduce – redukce
        - zkombinování výstupů z mapování
    - jednoúčelová agregace
      - pro dokumenty v jedné kolekci



# OPAKOVÁNÍ

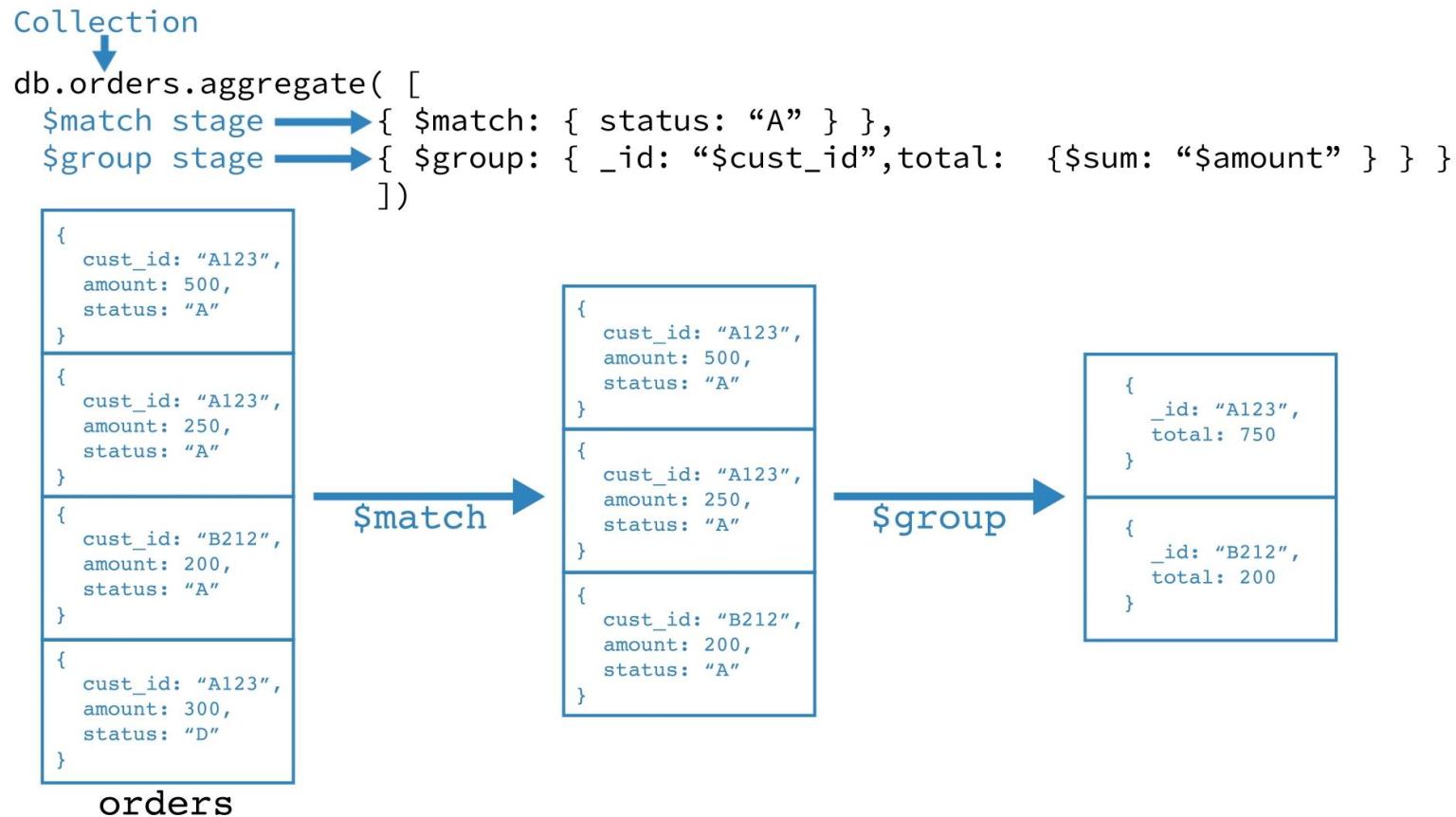
- **agregační roura**
  - založena na principu roury na zpracování dat
  - dokumenty vstupují do vícefázové roury, která je transformuje na agregovaný výsledek
    - každá fáze transformuje dokumenty pro další fázi roury
    - ne každá fáze vytváří pro každý vstupní dokument výstupní dokument
      - některé např. dokumenty filtroují
    - většina fází se může i opakovat
    - po průchodu celou rourou je získán výsledek

```
db.orders.aggregate([
  { $match: { status: "A" } },
  { $group: { _id: "$cust_id", total: { $sum: "$amount" } } }
])
```

<https://docs.mongodb.com/manual/aggregation/>



# OPAKOVÁNÍ



<https://docs.mongodb.com/manual/aggregation/>



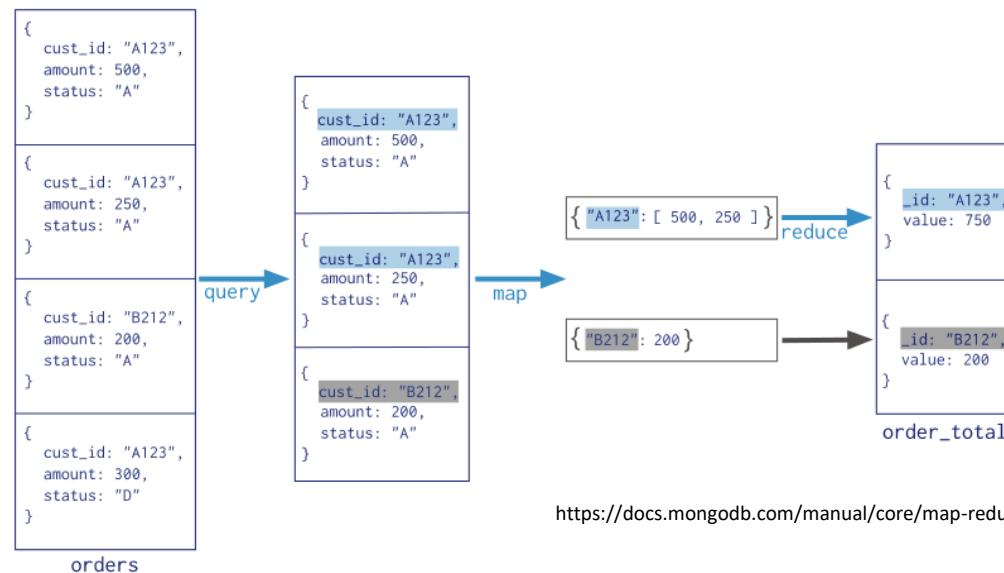
# OPAKOVÁNÍ

- map-reduce
  - paradigma paralelního zpracování dat
    - pro převedení velkého množství dat na užitečné agregované výsledky
    - zpracování je rozdělené na dvě základní operace
      - map – data jsou rozdělena na chunky a mohou být paralelně zpracována
      - reduce – slučuje výsledky z operace map do finálního výsledku
  - paralelní výpočty na obrovských datech v clusterech
  - MongoDB
    - využívá JavaScript funkcí pro operace *map*, *reduce* a volitelně *finalize*
    - operace *map*
      - mapuje hodnoty ke klíči
      - pokud pro daný klíč existuje více hodnot, mapovány budou všechny
      - funkce *emit(klíč, hodnota)* vrací výstupní dokumenty spojující klíče s hodnotami
    - operace *reduce*
      - bere klíč a agreguje k němu namapované hodnoty
      - jako vstup bere výstup z operace *map*, který zpracovává
    - volitelně operace *finalize*



# OPAKOVÁNÍ

```
Collection
↓
db.orders.mapReduce(
  map → function() { emit( this.cust_id, this.amount ); },
  reduce → function(key, values) { return Array.sum( values ) },
  query → { query: { status: "A" } },
  output → { out: "order_totals" }
)
```



<https://docs.mongodb.com/manual/core/map-reduce/>



# ČÁST II.: SEARCH-ENGINE DATABÁZE



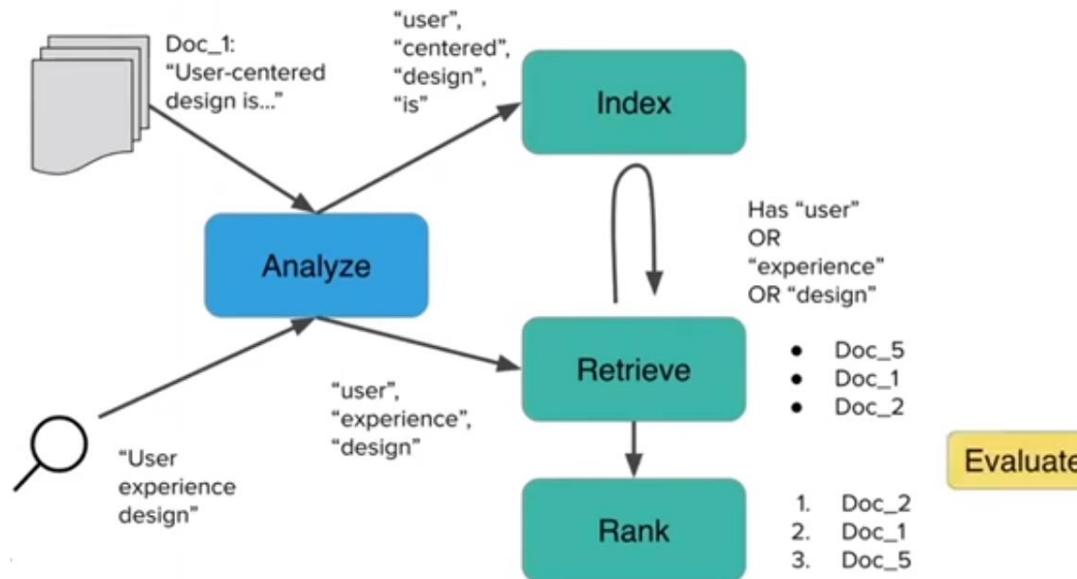
# SEARCH-ENGINE DATABÁZE

- NoSQL dokumentové databáze zaměřené na vyhledání obsahu
- využívají indexování
  - kategorizace podobných vlastností mezi daty
  - urychlení vyhledávání
- optimalizované pro práci s daty
  - velké množství dat
  - strukturovaná i nestrukturovaná data
    - volné schéma
- poskytují speciální funkce
  - full-textové vyhledávání
  - složité vyhledávací výrazy
  - řazení výsledků



<https://www.signitysolutions.com/enterprise-search-implementation>

# SEARCH-ENGINE DATABÁZE



<https://www.youtube.com/watch?v=dqRDyeFJuvk>

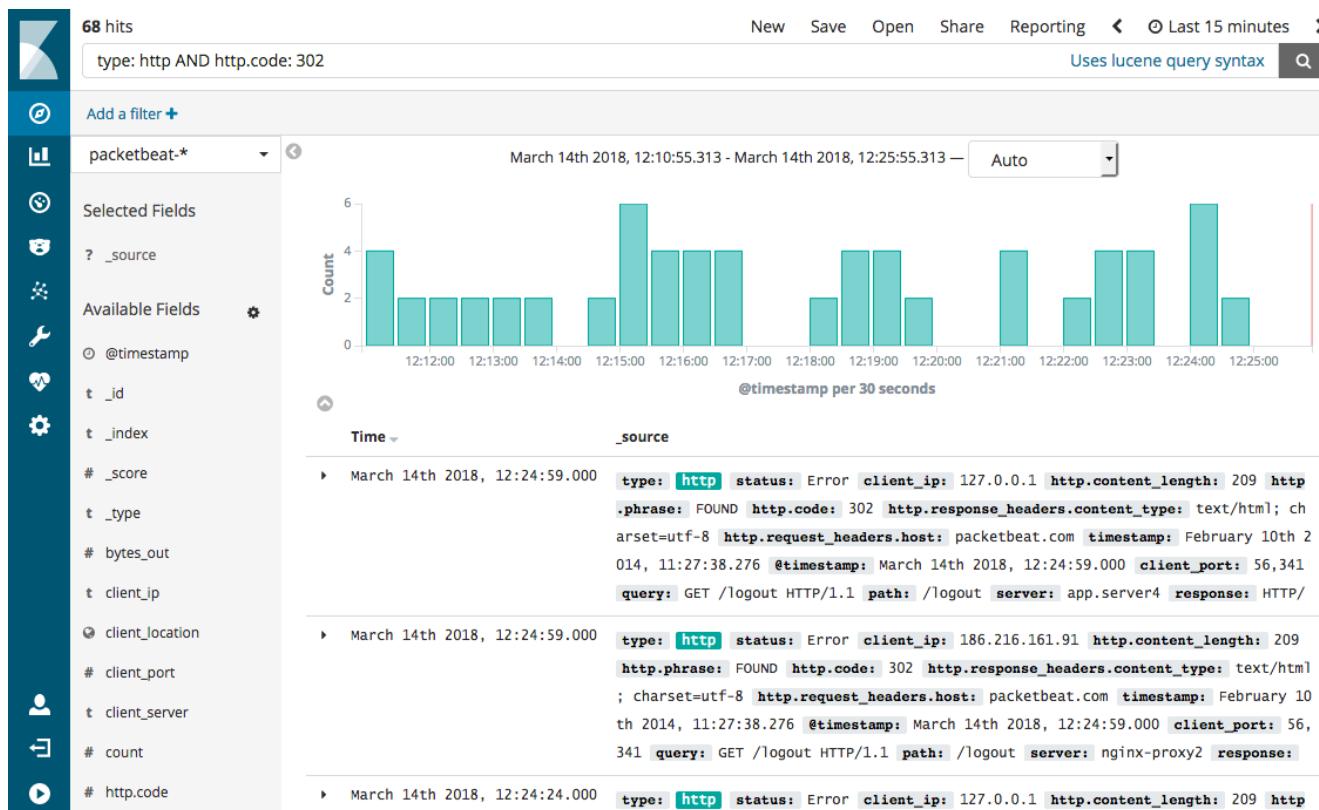
# SEARCH-ENGINE DATABÁZE

- distribuované vyhledávání
  - škálování, sharding, replikace
- příklady aplikace
  - textové vyhledávání
    - elektronický obchod
    - automatické dokončování hledaných výrazů, doporučení
    - setřídění výsledků na základě různých kritérií (jméno, cena, datum vydání, ...)
  - logování a analýza
    - centralizace a zaindexování logů z různých zařízení
- nejvýznamnější zástupci
  - Elasticsearch, OpenSearch
  - Solr, Splunk, MarkLogic, Algolia





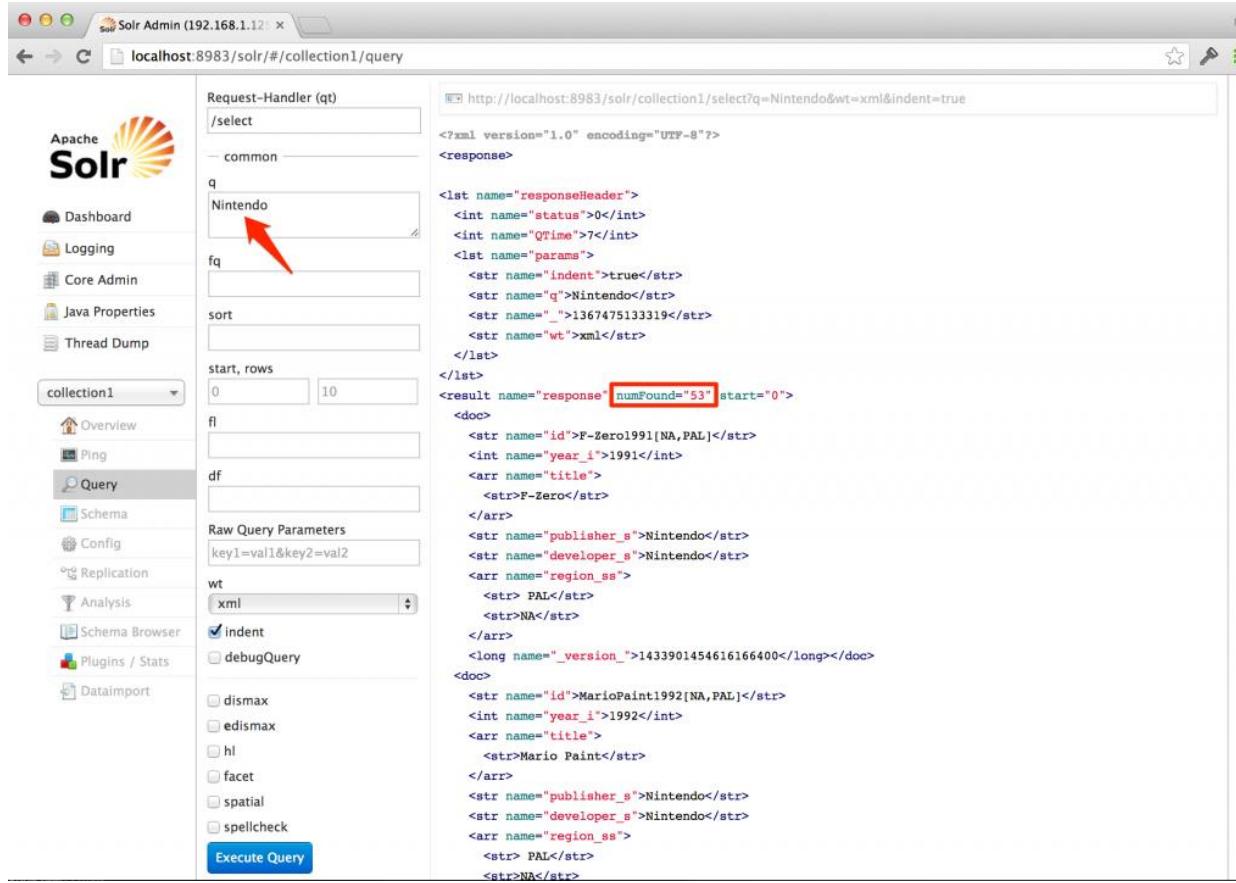
# SEARCH-ENGINE DATABÁZE



<https://www.elastic.co/guide/en/beats/packetbeat/current/kibana-queries-filters.html>



# SEARCH-ENGINE DATABÁZE



The screenshot shows the Apache Solr Admin interface running on localhost:8983. The left sidebar has 'collection1' selected under 'Overview'. The main area shows a search form with the following parameters:

- Request-Handler (qt): /select
- q: Nintendo (highlighted with a red arrow)
- fq: (empty)
- sort: (empty)
- start, rows: 0, 10
- fl: (empty)
- df: (empty)
- Raw Query Parameters: key1=val&key2=val2
- wt: xml
- Indent: checked
- debugQuery: unchecked
- dismax: unchecked
- edismax: unchecked
- hl: unchecked
- facet: unchecked
- spatial: unchecked
- spellcheck: unchecked

Below the form is a large text area displaying the XML response from the search query. The response shows a total of 53 results found, with two document snippets displayed:

```
<?xml version="1.0" encoding="UTF-8"?>
<response>
<lst name="responseHeader">
<int name="status">0</int>
<int name="QTime">7</int>
<lst name="params">
<str name="indent">true</str>
<str name="q">Nintendo</str>
<str name="">1367475133319</str>
<str name="wt">xml</str>
</lst>
</list>
<result name="response" numFound="53" start="0">
<doc>
<str name="id">F-Zero1991[NA,PAL]</str>
<int name="year_i">1991</int>
<arr name="title">
<str>F-Zero</str>
</arr>
<str name="publisher_s">Nintendo</str>
<str name="developer_s">Nintendo</str>
<arr name="region_ss">
<str>PAL</str>
<str>NA</str>
</arr>
<long name="_version_">1433901454616166400</long>
</doc>
<doc>
<str name="id">MarioPaint1992[NA,PAL]</str>
<int name="year_i">1992</int>
<arr name="title">
<str>Mario Paint</str>
</arr>
<str name="publisher_s">Nintendo</str>
<str name="developer_s">Nintendo</str>
<arr name="region_ss">
<str>PAL</str>
<str>NA</str>
</arr>
</doc>

```

<https://www.blendinteractive.com/news/searching-with-solr/>



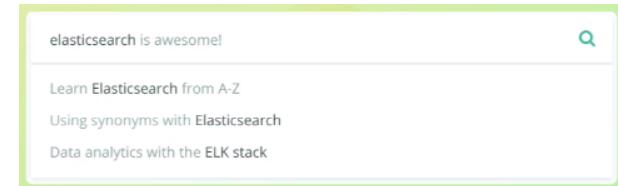


# ČÁST III.: ELASTICSEARCH



# ELASTICSEARCH

- engine pro full-textové prohledávání a analýzu
  - open source
    - napsán v Javě
    - založený na Apache Lucene
  - dokumentová databáze
    - data ukládána do dokumentů s polí
    - využívá JSON
  - dotazování pomocí REST API
  - distribuovaný
    - vysoká škálovatelnost
    - rychlosť prohledávání



elasticsearch is awesome!

Learn Elasticsearch from A-Z

Using synonyms with Elasticsearch

Data analytics with the ELK stack

<https://www.udemy.com/course/elasticsearch-complete-guide/>



NETFLIX

Uber

slack

twilio

Microsoft

# ELASTICSEARCH

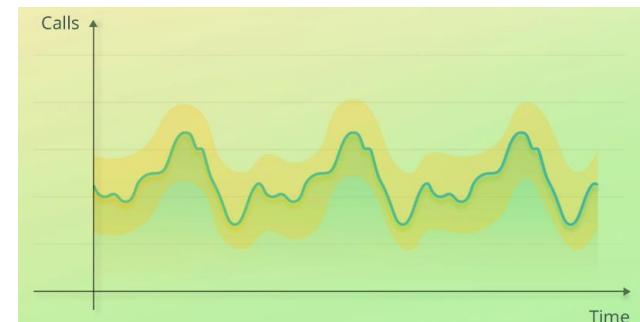
- v aktivním vývoji
  - Elastic NV
  - poslední verze 8.13.2
    - 04/24
- součástí Elastic Stack



<https://infohub.delltechnologies.com/l/elastic-stack-on-dell-emc-vxrail/elastic-stack-10>

# ELASTICSEARCH

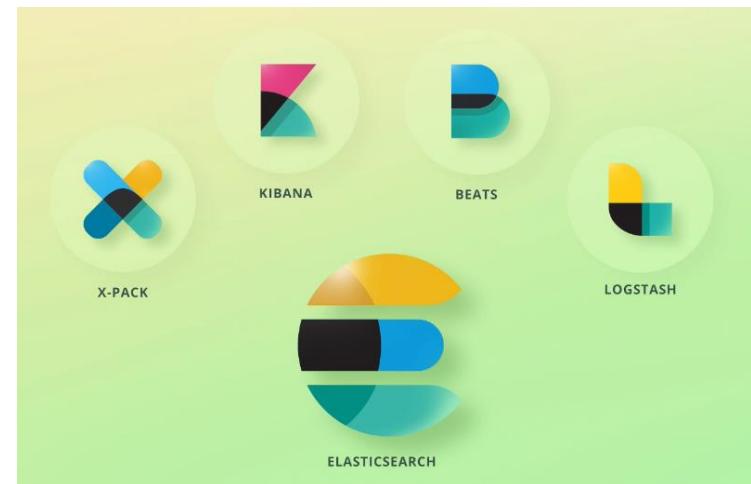
- možnosti aplikace
  - komplexní prohledávač pro online aplikace
    - prohledávání různých dat
      - produkty, kategorie, příspěvky na blogu, ...
    - automatické doplňování, autokorekce, zvýrazňování shod, synonyma, ...
    - řazení nalezených výsledků
      - co zobrazit první?
      - např. podle hodnocení
      - filtrování uživatelem podle ceny, značky, barvy, kategorie, ...
- platforma pro analýzu dat
  - agregování dat a následná tvorba grafů
  - např. analýza logů z různých serverů
    - s možností upozornění na základě chyb, spotřeby, ...
    - Application Performance Management
  - analýza velkého množství dat
    - možnost využití strojového učení pro predikce
    - detekce anomalií



<https://www.udemy.com/course/elasticsearch-complete-guide/>

# ELASTIC STACK

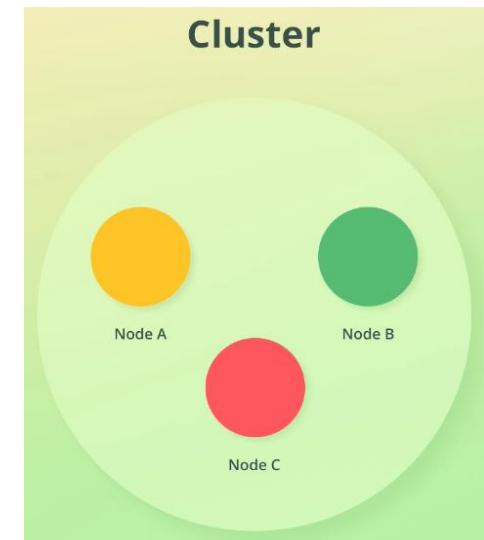
- technologie vyvinuté a spravované Elastic BV
  - silná synergie mezi technologiemi
    - často používané společně s Elasticsearch
  - Kibana
    - platforma pro analýzu a vizualizaci dat
  - Logstash
    - roura (pipeline) na zpracování dat
  - X-Pack
    - balík funkcí do Elasticsearch a Kibany
  - Beats
    - kolekce data shippers



<https://www.udemy.com/course/elasticsearch-complete-guide/>

# ARCHITEKTURA ELASTICSEARCH

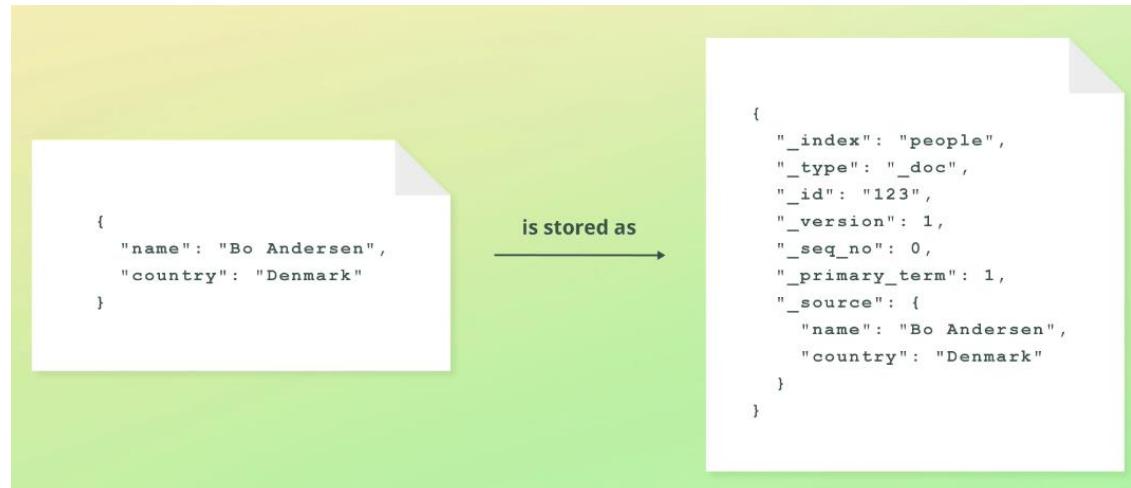
- základem je **uzel** (node)
  - automaticky založen při spuštění Elasticsearch
  - instance Elasticsearch obsahující data
  - uzlů může být samozřejmě více
    - každý uzel ukládá část dat
      - škálování
    - na jednom stroji může být více uzlů
    - fyzické, virtuální, docker kontejner, ...
- každý uzel je součástí **clusteru**
  - automaticky založen při vytvoření prvního uzlu
    - uzel je buď přiřazen do existujícího clusteru
    - nebo uzel svým vznikem zakládá nový cluster
  - může jich být také více
    - ale většinou pro logicky oddělené úlohy



<https://www.udemy.com/course/elasticsearch-complete-guide/>

# ARCHITEKTURA

- základní jednotkou pro ukládání dat je **dokument**
  - JSON objekty obsahující data
  - Elasticsearch ukládá společně s metadaty
    - původní dokument v poli \_source



<https://www.udemy.com/course/elasticsearch-complete-guide/>

# ARCHITEKTURA

- data jsou organizována v **indexech**
  - každý dokument uložen v indexu
    - neomezený počet dokumentů
  - logicky shlukují dokumenty
    - kolekce dokumentů s podobnými vlastnostmi
    - poskytují možnosti pro škálování a dostupnost
  - dotazy jsou spouštěny nad indexy



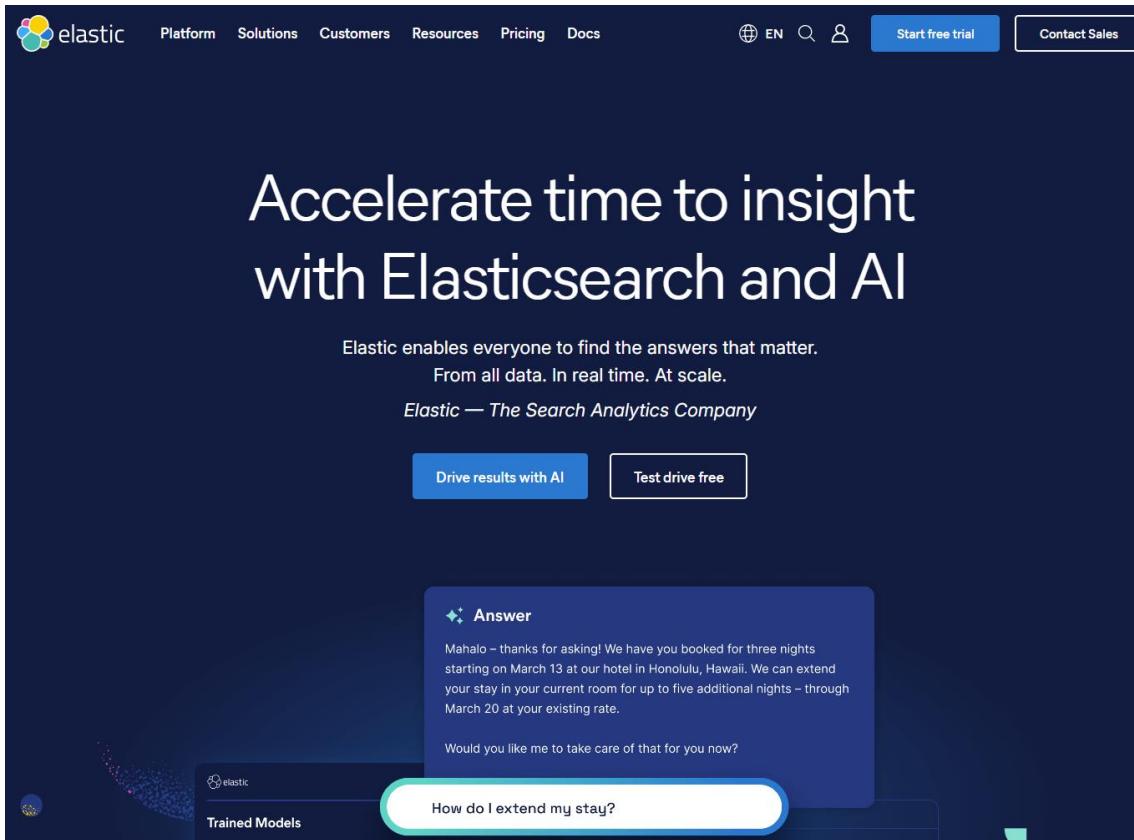
<https://www.udemy.com/course/elasticsearch-complete-guide/>



# ČÁST IV.: ELASTICSEARCH PRAKTICKY



# ELASTICSEARCH



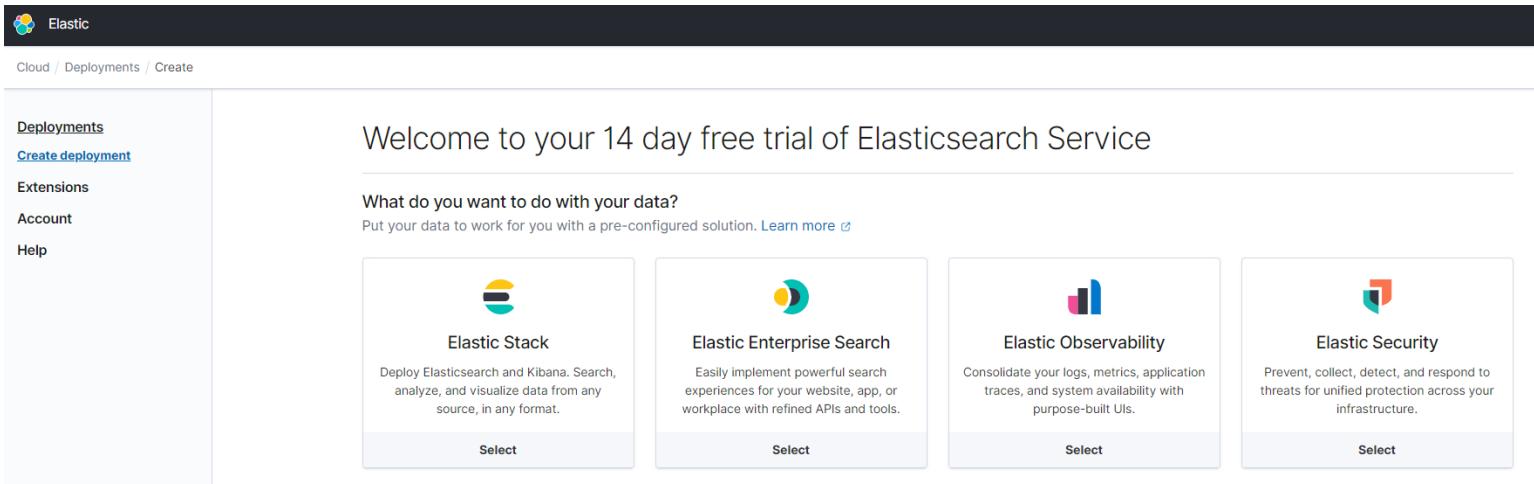
The screenshot shows the official Elasticsearch website homepage. At the top, there's a navigation bar with links for Platform, Solutions, Customers, Resources, Pricing, and Docs. On the right side of the bar are icons for language (EN), search, user profile, and buttons for "Start free trial" and "Contact Sales". The main headline reads "Accelerate time to insight with Elasticsearch and AI". Below it, a sub-headline states "Elastic enables everyone to find the answers that matter. From all data. In real time. At scale." followed by the company tagline "Elastic — The Search Analytics Company". Two prominent buttons are visible: "Drive results with AI" (blue background) and "Test drive free" (white background). A large blue callout box in the center contains an "Answer" section with a snippet of text about hotel bookings and a question "Would you like me to take care of that for you now?". At the bottom left, there's a "Trained Models" section with a "How do I extend my stay?" button.

<https://www.elastic.co/>



# ELASTICSEARCH

- Elastic Cloud
  - hostované a spravované řešení pro snadné nasazení Elasticsearch
  - na 14 dní zdarma
    - + 7 dní prodloužení
  - placené možnosti



The screenshot shows the Elastic Cloud interface for creating a new deployment. The top navigation bar includes the Elastic logo, a search bar, and links for 'Cloud / Deployments / Create'. On the left, a sidebar menu lists 'Deployments' (selected), 'Create deployment', 'Extensions', 'Account', and 'Help'. The main content area features a large 'Welcome to your 14 day free trial of Elasticsearch Service' message. Below it, a section asks 'What do you want to do with your data?' with a note to 'Put your data to work for you with a pre-configured solution.' A 'Learn more' link is provided. Four service options are listed in cards: 'Elastic Stack' (with a 'Select' button), 'Elastic Enterprise Search' (with a 'Select' button), 'Elastic Observability' (with a 'Select' button), and 'Elastic Security' (with a 'Select' button).



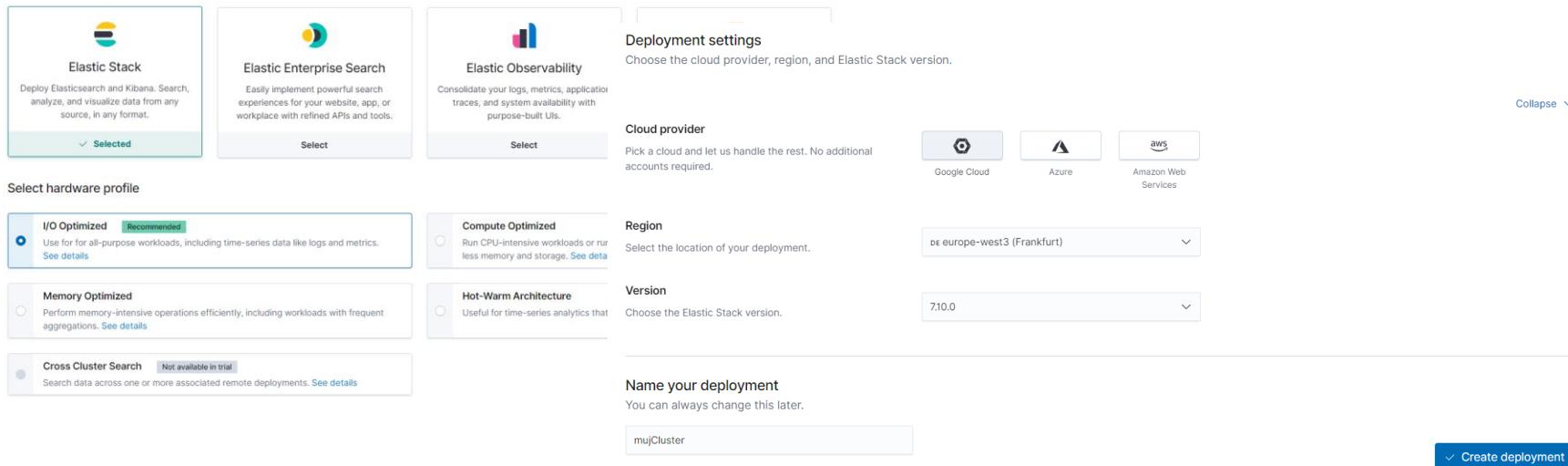
# ELASTICSEARCH

- Elastic Cloud
  - vytvoření clusteru (deployment)
    - výběr nástrojů, optimalizace, provozovatele, regionu a jména clusteru
    - vytvoří uživatelské jméno a heslo pro přístup k Elasticsearch a Kibana

Welcome to your 14 day free trial of Elasticsearch Service

What do you want to do with your data?

Put your data to work for you with a pre-configured solution. [Learn more](#)



The screenshot shows the configuration interface for a 14-day free trial of the Elasticsearch Service. It includes sections for selecting a hardware profile (I/O Optimized, Recommended), deployment settings (Cloud provider: Google Cloud, Azure, AWS), and naming the deployment (mujCluster). A 'Create deployment' button is at the bottom right.

Elastic Stack

Elastic Enterprise Search

Elastic Observability

Deployment settings

Cloud provider

Select hardware profile

I/O Optimized (Recommended)

Compute Optimized

Memory Optimized

Hot-Warm Architecture

Cross Cluster Search

Name your deployment

Create deployment





# ELASTICSEARCH

- **Elastic Cloud**

## Deployments

Deployment name: mujCluster Deployment status: Healthy

Deployment ID: a6fc3e3 Deployment version: v7.10.0

Cloud ID: mujCluster:ZXVyb3B1LXd1c30zLmdjccC5jbG91ZC51cy5pbvQ10GE0NThi0TUwZTU0YzQwYTQzMWU0ZDgwOTE3Yz1lMyQ3YjBmMzE30TQ5N2U0MGZ10DFKNjVKNJ1MTBhY2Jh0A==

Open Kibana | Manage

Applications: Elasticsearch / Copy endpoint, Kibana / Launch / Copy endpoint, APM / Launch / Copy endpoint

Instances

Zone	Instance	Status	Disk allocation	JVM memory pressure
europe-west3-a	Tiebreaker #2	Healthy	0 GB / 2 GB	9%
europe-west3-b	Instance #0	Healthy	23 MB / 120 GB	0%
europe-west3-c	Instance #0	Healthy	22 MB / 120 GB	0%
europe-west3-c	Instance #1	Healthy	22 MB / 120 GB	4%
europe-west3-c	Instance #1	Healthy	21 MB / 120 GB	6%



# ELASTICSEARCH

- kromě Elastic Cloud samozřejmě i lokální instalace
  - na všechny moderní platformy
    - [Elasticsearch](#)
    - [Kibana](#)

## Download Elasticsearch

### 1 Download and unzip Elasticsearch

Choose platform:

Windows

 Windows

 sha  asc



# ELASTICSEARCH

- kromě Elastic Cloud samozřejmě i lokální instalace
  - spuštění přes bin/elasticsearch.bat
  - Kibana přes bin/kibana.bat    ->    <http://localhost:5601>

```
D:\Programy\elasticsearch-7.10.0>bin\elasticsearch.bat
[2020-11-13T15:08:20,250][INFO ][o.e.n.Node] [1SHARK1-PC] version[7.10.0], pid[1048], build[default/zip/51e9d6f22758d0374aaef3f5c6e8f3a7997850f96/2020-11-09T21:30:33.964949Z], OS[Windows 10/10.0/amd64], JVM[AdoptOpenJDK/OpenJDK 64-Bit Server VM/15.0.1/15.0.1+9]
[2020-11-13T15:08:20,254][INFO ][o.e.n.Node] [1SHARK1-PC] JVM home [D:\Programy\elasticsearch-7.10.0\jdk], using bundled JDK [true]
[2020-11-13T15:08:20,267][INFO ][o.e.n.Node] [1SHARK1-PC] JVM arguments [-Des.networkaddress.cache.ttl=60, -Des.networkaddress.cache.negative.ttl=10, -XX:+AlwaysPreTouch, -Xss1m, -Djava.awt.headless=true, -Dfile.encoding=UTF-8, -Djna.nosys=true, -XX:-OmitStackTraceInFastThrow, -XX:+ShowCodeDetailsInExceptionMessages, -Dio.netty.noUnsafe=true, -Dio.netty.noKeySetOptimization=true, -Dio.nettyallocator.numDirectArenas=0, -Dlog4j.shutdownHookEnabled=false, -Dlog4j2.disable.jmx=true, -Djava.locale.providers=SPI,COMPAT, -Xms1g, -Xmx1g, -XX:+UseG1GC, -XX:G1ReservePercent=25, -XX:InitiatingHeapOccupancyPercent=30, -Djava.io.tmpdir=C:\Users\Rimmer\AppData\Local\Temp\elasticsearch, -XX:HeapDumpOnOutOfMemoryError, -XX:HeapDumpPath=data, -XX:ErrorFile=logs/hs_err_pid%p.log, -Xlog:gc*,gc+age+trace,safepoint:file=log/s/gc.log:utctime,pid,tags:filecount=32,filesize=64m, -XX:MaxDirectMemorySize=536870912, -Delasticsearch, -Des.path.home=D:\Programy\elasticsearch-7.10.0, -Des.path.conf=D:\Programy\elasticsearch-7.10.0\config, -Des.distribution.flavor=default, -Desdistribution.type=zip, -Des.bundled.jdk=true]
[2020-11-13T15:08:26,798][INFO ][o.e.p.PluginsService] [1SHARK1-PC] loaded module [aggs-matrix-stats]
[2020-11-13T15:08:26,799][INFO ][o.e.p.PluginsService] [1SHARK1-PC] loaded module [analysis-common]
[2020-11-13T15:08:26,799][INFO ][o.e.p.PluginsService] [1SHARK1-PC] loaded module [constant-keyword]
[2020-11-13T15:08:26,799][INFO ][o.e.p.PluginsService] [1SHARK1-PC] loaded module [flattened]
[2020-11-13T15:08:26,800][INFO ][o.e.p.PluginsService] [1SHARK1-PC] loaded module [frozen-indices]
[2020-11-13T15:08:26,800][INFO ][o.e.p.PluginsService] [1SHARK1-PC] loaded module [ingest-common]
[2020-11-13T15:08:26,800][INFO ][o.e.p.PluginsService] [1SHARK1-PC] loaded module [ingest-geoip]
[2020-11-13T15:08:26,801][INFO ][o.e.p.PluginsService] [1SHARK1-PC] loaded module [ingest-user-agent]
[2020-11-13T15:08:26,801][INFO ][o.e.p.PluginsService] [1SHARK1-PC] loaded module [kibana]
[2020-11-13T15:08:26,801][INFO ][o.e.p.PluginsService] [1SHARK1-PC] loaded module [lang-expression]
[2020-11-13T15:08:26,802][INFO ][o.e.p.PluginsService] [1SHARK1-PC] loaded log [14:39:33.005] [info][plugins][taskManager][taskManager] TaskManager is identified by the Kibana UUID: 3e304fca-ef3e-493c-aba7-680bc8c6c00d
[2020-11-13T15:08:26,802][INFO ][o.e.p.PluginsService] [1SHARK1-PC] loaded log [14:39:33.236] [info][crossClusterReplication][plugins] Your basic license does not support crossClusterReplication. Please upgrade your license.
[2020-11-13T15:08:26,803][INFO ][o.e.p.PluginsService] [1SHARK1-PC] loaded log [14:39:33.250] [info][plugins][watcher] Your basic license does not support watcher. Please upgrade your license.
[2020-11-13T15:08:26,804][INFO ][o.e.p.PluginsService] [1SHARK1-PC] loaded log [14:39:33.252] [info][kibana-monitoring][monitoring][plugins]
[2020-11-13T15:08:26,804][INFO ][o.e.p.PluginsService] [1SHARK1-PC] loaded Starting monitoring stats collection
[2020-11-13T15:08:26,805][INFO ][o.e.p.PluginsService] [1SHARK1-PC] loaded log [14:39:36.358] [info][listening] Server running at http://localhost:5601
[2020-11-13T15:08:26,805][INFO ][o.e.p.PluginsService] [1SHARK1-PC] loaded log [14:39:36.957] [info][server][Kibana][http] http server running at http://localhost:5601
[2020-11-13T15:08:26,805][INFO ][o.e.p.PluginsService] [1SHARK1-PC] loaded log [14:39:37.000] [info][server][Kibana][http] http server stopped at http://localhost:5601
[2020-11-13T15:08:26,806][INFO ][o.e.p.PluginsService] [1SHARK1-PC] loaded module [unsigned-long]
[2020-11-13T15:08:26,806][INFO ][o.e.p.PluginsService] [1SHARK1-PC] loaded module [vectors]
[2020-11-13T15:08:26,807][INFO ][o.e.p.PluginsService] [1SHARK1-PC] loaded module [wildcard]
[2020-11-13T15:08:26,807][INFO ][o.e.p.PluginsService] [1SHARK1-PC] loaded module [x-pack-analytics]
```



# ELASTICSEARCH

- od verze 8.0 je potřeba nastavit hesla pro uživatele
  - bin/elasticsearch-reset-password -i -u elastic
  - v základu uživatel elastic

```
D:\Programy\elasticsearch-8.1.2>bin\elasticsearch-reset-password -i -u elastic
warning: ignoring JAVA_HOME=C:\Progra~1\Java\jdk-16; using bundled JDK
This tool will reset the password of the [elastic] user.
You will be prompted to enter the password.
Please confirm that you would like to continue [y/N]y

Enter password for [elastic]:
Re-enter password for [elastic]:
Password for the [elastic] user successfully reset.
```

```
D:\>curl http://localhost:9200
{
  "name" : "1SHARK1-PC",
  "cluster_name" : "elasticsearch",
  "cluster_uuid" : "MBaHUKXLRLK-G0GTQN4-oQ",
  "version" : {
    "number" : "7.10.0",
    "build_flavor" : "default",
    "build_type" : "zip",
    "build_hash" : "51e9d6f22758d0374a0f3f5c6e8f3a7997850f96",
    "build_date" : "2020-11-09T21:30:33.964949Z",
    "build_snapshot" : false,
    "lucene_version" : "8.7.0",
    "minimum_wire_compatibility_version" : "6.8.0",
    "minimum_index_compatibility_version" : "6.0.0-beta1"
  },
  "tagline" : "You Know, for Search"
```



# KIBANA

- od verze 8.0 potřeba nastavit Kibantu kvůli zabezpečení
  - potřeba zadat enrollment token
  - bin/elasticsearch-create-enrollment-token.bat --scope kibana

```
D:\Programy\elasticsearch-8.1.2>bin\elasticsearch-create-enrollment-token.bat --scope kibana
```

- následně ověřit kód vyspaný v konzoli Kibany

```
i Kibana has not been configured.  
Go to http://localhost:5601/?code=008964 to get started.  
  
Your verification code is: 008 964
```

- a přihlásit vybraným uživatelem (např. elastic)



# PRVNÍ DOTAZY

- komunikace s Elasticsearch REST API
  - využití HTTP žádostí pro přístup k datům
    - metody POST, GET, PUT a DELETE (odpovídá CRUD)
    - libovolný HTTP klient
  - využití Kibany
    - vestavěná konzole
    - překládá uživatelské dotazy na HTTP žádosti
    - žádosti následně posílány do Elasticsearch
    - dotaz na stav clusteru
      - GET /\_cluster/health
        - první část
          - žádost – GET – získání informací
        - druhá část
          - cesta – reálná část přidávána automaticky Kibanou
          - \_cluster – specifikuje API (obecně začínají \_), ke kterému se přistupuje
          - health – samotný příkaz





# PRVNÍ DOTAZY

- komunikace s Elasticsearch REST API
  - využití Kibany
    - vypsání uzlů

GET /\_cat/nodes?v

- \_cat – API pro výpis dat ve formátu čitelném pro lidi
- ?v – dotazovací parametr zajišťující vypsání hlavičky výstupu

The screenshot shows the Elasticsearch Dev Tools interface with two requests displayed:

- Request 1: GET /\_cat/nodes?v**  
Response status: 200 - OK | 45 ms  
Result:

ip	heap.percent	ram.percent	cpu	load_1m	load_5m	load_15m	node.role	master	name
127.0.0.1	58	38	3				cdhilmrstw *	1	SHARK1-PC
- Request 2: GET /\_cat/indices?v**  
Response status: 200 - OK | 24 ms  
Result:

health	status	index	uuid	pri	rep	docs.count	docs.deleted	store.size	pri.store.size
green	open	.apm-custom-link	dyIKFNC4QjmRBOqLpChEeA	1	0	0	0	208b	208b
green	open	.kibana_task_manager_1	wlrbdBmTE2GLAi6QVxHia	1	0	5	22	117.3kb	117.3kb
green	open	.apm-agent-configuration	b7HIVJ4irKWSU7wve5f0UA	1	0	0	0	208b	208b
green	open	.kibana-event-log-7.10.0-000001	Yff0wU_kRqlpxKesSWC2SA	1	0	2	0	10.9kb	10.9kb
green	open	.kibana_1	DIEx39orSt-7_bAEYnL3HQ	1	0	18	1	10.4mb	10.4mb



# PRVNÍ DOTAZY

- komunikace s Elasticsearch REST API

- využití cURL
  - od verze 8.0 nutná autentifikace
  - nutno zadat celou cestu
  - nutno předat uživatele s heslem
  - nutno předat HTTP certifikát
  - předání certifikátu je ale v posledních verzích zabugované
  - problém na MS Windows
  - použití parametru `-k` pro ignoranci

```
D:\Programy\elasticsearch-8.1.2>curl -k --cacert ./config/certs/http_ca.crt -u elastic https://localhost:9200
Enter host password for user 'elastic':
```



# PRVNÍ DOTAZY

- komunikace s Elasticsearch REST API

- využití cURL
  - od verze 8.0 nutná autentifikace
  - opět potřeba celá cesta, uživatel a certifikát
  - složitější dotaz
    - základní vyhledávání
    - nutno specifikovat Content-Type v případě těla dotazu
      - JSON, ale podpora pro např. i YAML

```
History Settings Help
1 GET /.kibana/_search
2 {
3   "query" : {
4     | "match_all": {}
5   }
6 }
```

```
D:\>curl -XGET "http://localhost:9200/_search" -H "Content-Type: application/json" -d"{"query": {"match_all": {}}}"
```

- a co Elastic Cloud?
  - nutno správně nastavit endpoint – umístění clusteru

```
D:\>curl -XGET -u elastic:xa5MvFmJ8vb1BI6VWwVHNugE "https://58a458b950e54c40a411e4d80917c9e3.europe-west3.gcp.cloud.es.io:9243/_cluster/health"
{"cluster_name":"58a458b950e54c40a411e4d80917c9e3","status":"green","timed_out":false,"number_of_nodes":3,"number_of_data_nodes":2,"active_primary_shards":16,"active_shards":32,"relocating_shards":0,"initializing_shards":0,"unassigned_shards":0,"delayed_unassigned_shards":0,"number_of_pending_tasks":0,"number_of_in_flight_fetch":0,"task_max_waiting_in_queue_millis":0,"active_shards_percent_as_number":100.0}
```





# DOTAZY

- pro veškeré následující ukázky bude využívána Kibana
  - z důvodu usnadnění
  - volba komunikačního klienta je na vás
  - ukázky vychází z Udemy kurzu Complete Guide to Elasticsearch
    - <https://www.udemy.com/course/elasticsearch-complete-guide>



# SPRÁVA DOKUMENTŮ

- vytvoření indexu

PUT /products

```
1 PUT /products ▶ 🔍
1 ↴ { "acknowledged" : true,
2   "shards_acknowledged" : true,
3   "index" : "products"
4 ↵ }
```

- smazání indexu

DELETE /products

- indexování dokumentů

- vložení dokumentu do indexu
- nutná specifikace indexu
- dokument ve formátu JSON

POST /products/\_doc

```
{ "name" : "Coffee Maker",
  "price" : 64,
  "in_stock" : 10 }
```

```
1 PUT /products
2
3 POST /products/_doc ▶ 🔍
4 ↴ { "name" : "Coffee Maker",
5   "price" : 64,
6   "in_stock" : 10
7 ↵ }
```

```
1 ↴ { "_index" : "products",
2   "_type" : "_doc",
3   "_id" : "f_0x23UBkaLlnoKa18am",
4   "_version" : 1,
5   "result" : "created",
6   "_shards" : {
7     "total" : 2,
8     "successful" : 2,
9     "failed" : 0
10   },
11   "_seq_no" : 1,
12   "_primary_term" : 1
13 ↵ }
```



# SPRÁVA DOKUMENTŮ

- indexování dokumentů
  - \_id – automaticky vytvořený jednoznačný identifikátor
    - je možné specifikovat sám
  - \_shards – informace o shardování
  - \_primary\_term – počet změn primárního shardu
  - \_seq\_no – počet operací s dokumentem
    - umožňují obnovu v případě selhání primárního shardu

```
1 PUT /products
2
3 POST /products/_doc 
4 {
5   "name" : "Coffee Maker",
6   "price" : 64,
7   "in_stock" : 10
8 }
```

```
1 {
2   "_index" : "products",
3   "_type" : "_doc",
4   "_id" : "f_0x23UBkaLlnoKa18am",
5   "_version" : 1,
6   "result" : "created",
7   "_shards" : {
8     "total" : 2,
9     "successful" : 2,
10    "failed" : 0
11  },
12  "_seq_no" : 1,
13  "_primary_term" : 1
14 }
```

```
1 PUT /products
2
3 PUT /products/_doc/100 
4 {
5   "name" : "Toaster",
6   "price" : 49,
7   "in_stock" : 4
8 }
```

```
1 {
2   "_index" : "products",
3   "_type" : "_doc",
4   "_id" : "100",
5   "_version" : 1,
6   "result" : "created",
7   "_shards" : {
8     "total" : 2,
9     "successful" : 2,
10    "failed" : 0
11  },
12  "_seq_no" : 2,
13  "_primary_term" : 1
14 }
```





# SPRÁVA DOKUMENTŮ

- čtení dokumentu podle \_id

GET /products/\_doc/100

- pole found určuje, jestli byl dokument nalezen
- nalezený dokument je v poli \_source



```
1 GET /products/_doc/100 ▶ 🔍
1  {
2     "_index": "products",
3     "_type": "_doc",
4     "_id": "100",
5     "_version": 1,
6     "_seq_no": 2,
7     "_primary_term": 1,
8     "found": true,
9     "_source": {
10        "name": "Toaster",
11        "price": 49,
12        "in_stock": 4
13     }
14 }
```



# SPRÁVA DOKUMENTŮ

- update dokumentu podle \_id

```
POST /products/_update/100
{
    "doc" : {
        "in_stock" : 3,
        "tags" : ["electronics"],
    }
}
```

- reálně nedochází k úpravě dokumentu, ale k jeho nahrazení
  - atomická operace
  - dokumenty jsou neupravitelné
  - o vše se postará Update API
    - načtení dokumentu
    - úprava polí
    - nahrazení dokumentu

```
1 POST /products/_update/100
2 {
3     "doc" : {
4         "in_stock" : 3,
5         "tags" : ["electronics"]
6     }
7 }
8
9 GET /products/_doc/100
```

```
1 {
2     "_index" : "products",
3     "_type" : "_doc",
4     "_id" : "100",
5     "_version" : 3,
6     "result" : "noop",
7     "_shards" : {
8         "total" : 0,
9         "successful" : 0,
10        "failed" : 0
11    },
12    "_seq_no" : 4,
13    "_primary_term" : 1
14 }
```

```
1 POST /products/_update/100
2 {
3     "doc" : {
4         "in_stock" : 3,
5         "tags" : ["electronics"]
6     }
7 }
8
9 GET /products/_doc/100
```

```
1 {
2     "_index" : "products",
3     "_type" : "_doc",
4     "_id" : "100",
5     "_version" : 3,
6     "_seq_no" : 4,
7     "_primary_term" : 1,
8     "found" : true,
9     "_source" : {
10         "name" : "Toaster",
11         "price" : 49,
12         "in_stock" : 3,
13         "tags" : [
14             "electronics"
15         ]
16     }
17 }
```

# SPRÁVA DOKUMENTŮ

- scripted update dokumentu podle \_id
  - možnost vlastní logiky využívající přístup k hodnotám dokumentu
    - např. snížení in\_stock o 1

```
1 POST /products/_update/100 | ➤🔍
2 { "script": {
3   | "source": "ctx._source.in_stock--"
4   }
5 }
6 }
7
8 GET /products/_doc/100
```

```
1 { "_index": "products",
2   "_type": "_doc",
3   "_id": "100",
4   "_version": 4,
5   "result": "updated",
6   "_shards": {
7     "total": 2,
8     "successful": 2,
9     "failed": 0
10    },
11   "_seq_no": 5,
12   "_primary_term": 1
13 }
```

```
1 { "_index": "products",
2   "_type": "_doc",
3   "_id": "100",
4   "_version": 4,
5   "_seq_no": 5,
6   "_primary_term": 1,
7   "found": true,
8   "source": {
9     "name": "Toaster",
10    "price": 49,
11    "in_stock": 2,
12    "tags": [
13      "electronics"
14    ]
15  }
16 }
17 }
```

- ctx (kontext) – proměnná
- \_source – přístup k dokumentu
  - vrací objekt
- in\_stock – přístup k poli in\_stock
- – snížení o 1



# SPRÁVA DOKUMENTŮ

- scripted update dokumentu podle \_id
  - využití parametrů předaných například z jiné aplikace
    - např. snížení in\_stock o N v závislosti na parametru
    - pozn. vhodné by bylo, aby in\_stock nekleslo pod 0

```
1 POST /products/_update/100
2 {
3   "script": {
4     "source": "ctx._source.in_stock -= params.quantity",
5     "params": {
6       "quantity": 4
7     }
8   }
9 }
10
11 GET /products/_doc/100
```

```
1 POST /products/_update/100
2 {
3   "script": {
4     "source": "ctx._source.in_stock -= params.quantity",
5     "params": {
6       "quantity": 4
7     }
8   }
9 }
10
11 GET /products/_doc/100
```

```
1 [
2   "_index": "products",
3   "_type": "_doc",
4   "_id": "100",
5   "_version": 5,
6   "_seq_no": 6,
7   "_primary_term": 1,
8   "found": true,
9   "_source": {
10     "name": "Toaster",
11     "price": 49,
12     "in_stock": -2,
13     "tags": [
14       "electronics"
15     ]
16   }
17 ]
```

```
1 [
2   "_index": "products",
3   "_type": "_doc",
4   "_id": "100",
5   "_version": 5,
6   "result": "updated",
7   "_shards": {
8     "total": 2,
9     "successful": 2,
10    "failed": 0
11  },
12  "_seq_no": 6,
13  "_primary_term": 1
14 ]
```



# SPRÁVA DOKUMENTŮ

- scripted update dokumentu podle \_id
  - snížení in\_stock o 1, pokud je nějaké zboží na skladě
  - "" značí víceřádkový skript
  - noop – žádná operace

```

1 POST /products/_update/100
2 {
3   "doc" : {
4     | "in_stock" : 15
5   }
6 }
7
8 POST /products/_update/100      ▶ ⚙
9 {
10  "script": {
11    "source" : """
12    | if (ctx._source.in_stock == 0) {
13    |   ctx.op = 'noop';
14    }
15    | ctx._source.in_stock--;
16
17  }
18 }
19
20 GET /products/_doc/100

```

```

1 [
2   "_index" : "products",
3   "_type" : "_doc",
4   "_id" : "100",
5   "_version" : 10,
6   "result" : "updated",
7   "_shards" : {
8     | "total" : 2,
9     | "successful" : 2,
10    | "failed" : 0
11  },
12  | "seq_no" : 11,
13  | "_primary_term" : 1
14 ]
15

```

```

1 POST /products/_update/100
2 {
3   "doc" : {
4     | "in_stock" : 15
5   }
6 }
7
8 POST /products/_update/100
9 {
10  "script": {
11    "source" : """
12    | if (ctx._source.in_stock == 0) {
13    |   ctx.op = 'noop';
14    }
15    | ctx._source.in_stock--;
16
17  }
18 }
19

```

```

1 POST /products/_update/100
2 {
3   "doc" : {
4     | "in_stock" : 1
5   }
6 }
7
8 POST /products/_update/100      ▶ ⚙
9 {
10  "script": {
11    "source" : """
12    | if (ctx._source.in_stock == 0) {
13    |   ctx.op = 'noop';
14    }
15    | ctx._source.in_stock--;
16
17  }
18 }
19
20 GET /products/_doc/100

```

```

1 [
2   "_index" : "products",
3   "_type" : "_doc",
4   "_id" : "100",
5   "_version" : 10,
6   "found" : true,
7   "_source" : {
8     | "name" : "Toaster",
9     | "price" : 49,
10    | "in_stock" : 14,
11    | "tags" : [
12      | "electronics"
13    ]
14  }
15 ]
16
17
18

```

```

1 [
2   "_index" : "products",
3   "_type" : "_doc",
4   "_id" : "100",
5   "_version" : 12,
6   "result" : "noop",
7   "_shards" : {
8     | "total" : 0,
9     | "successful" : 0,
10    | "failed" : 0
11  },
12  | "seq_no" : 13,
13  | "_primary_term" : 1
14 ]
15

```



# SPRÁVA DOKUMENTŮ

- update by query
  - update více dokumentů jedním dotazem na základě podmínky
  - snížení in\_stock u všech dokumentů o 1

```
1 POST /products/_update_by_query | ▶ 🔍
2 {
3   "script" : {
4     | "source" : "ctx._source.in_stock--"
5   },
6   "query" : {
7     | "match_all": {}
8   }
9 }

10
11 GET /products/_search
12 {
13   "query" : {
14     | "match_all": {}
15   }
16
17

18 ↴ { "took" : 241,
19   "timed_out" : false,
20   "total" : 2,
21   "updated" : 2,
22   "deleted" : 0,
23   "batches" : 1,
24   "version_conflicts" : 0,
25   "noops" : 0,
26   "retries" : {
27     | "bulk" : 0,
28     | "search" : 0
29   },
30   "throttled_millis" : 0,
31   "requests_per_second" : -1.0,
32   "throttled_until millis" : 0,
33   "failures" : [ ] }
```

- skriptovací část
  - "script" – stará se o dekrementaci
- vyhledávací část
  - "query" – vybírá všechny dokumenty



# SPRÁVA DOKUMENTŮ

- upsert dokumentu podle \_id
  - podmíněný update nebo vložení
    - podle toho, jestli dokument existuje nebo ne
      - pokud dokument existuje, je spuštěn skript
      - v opačném případě je indexován nový dokument

```

1 ▾ { _index": "products",
2   "_type": "_doc",
3   "_id": "101",
4   "_version": 2,
5   "_seq_no": 15,
6   "_primary_term": 1,
7   "found": true,
8   "source": {
9     "name": "Blender",
10    "price": 399,
11    "in_stock": 6
12  }
13 ▾ }
14 ▾ }
```

```

1 POST /products/_update/101 ⏎ ⓘ
2 ▾ {
3   "script": {
4     "source": "ctx._source.in_stock++"
5   },
6   "upsert": {
7     "name": "Blender",
8     "price": 399,
9     "in_stock": 5
10  }
11 ▾ }
12
13 GET /products/_doc/100
14
```

```

1 ▾ { _index": "products",
2   "_type": "_doc",
3   "_id": "101",
4   "_version": 1,
5   "_result": "created",
6   "_shards": {
7     "total": 2,
8     "successful": 2,
9     "failed": 0
10  },
11 ▾ },
12   "_seq_no": 14,
13   "_primary_term": 1
14 ▾ }
```

```

1 POST /products/_update/101 ⏎ ⓘ
2 ▾ {
3   "script": {
4     "source": "ctx._source.in_stock++"
5   },
6   "upsert": {
7     "name": "Blender",
8     "price": 399,
9     "in_stock": 5
10  }
11 ▾ }
12
13 GET /products/_doc/100
14
```



# SPRÁVA DOKUMENTŮ

- nahrazení dokumentu podle \_id
  - vložením dokumentu se stejným \_id dojde k přepsání původního
- smazání dokumentu podle \_id

```
DELETE /products/_doc/101
```



```
1 DELETE /products/_doc/100
2
3 GET /products/_doc/100
4
5
6
7
8
9
10
11
12
13
14
```

```
1 {
2   "_index": "products",
3   "_type": "_doc",
4   "_id": "100",
5   "_version": 13,
6   "result": "deleted",
7   "_shards": {
8     "total": 2,
9     "successful": 2,
10    "failed": 0
11  },
12   "_seq_no": 18,
13   "_primary_term": 1
14 }
```

```
1 {
2   "_index": "products",
3   "_type": "_doc",
4   "_id": "100",
5   "found": false
6 }
```



# SPRÁVA DOKUMENTŮ

- delete by query
  - smazání více dokumentů jedním dotazem na základě podmínky



```
1 POST /products/_delete_by_query
2 {
3   "query" : {
4     "match_all": {}
5   }
6 }
```

```
1 {
2   "took" : 198,
3   "timed_out" : false,
4   "total" : 2,
5   "deleted" : 2,
6   "batches" : 1,
7   "version_conflicts" : 0,
8   "noops" : 0,
9   "retries" : {
10     "bulk" : 0,
11     "search" : 0
12   },
13   "throttled_millis" : 0,
14   "requests_per_second" : -1.0,
15   "throttled_until_millis" : 0,
16   "failures" : [ ]
17 }
```



# SPRÁVA DOKUMENTŮ

- dávkové zpracování
  - umožňuje indexaci, update a smazání dokumentů jedním dotazem
  - zpracování jednotlivých dotazů v dávkách
- Bulk API
  - využívá data formátovaná v NDJSON

```
action_and_metadata \n
optional_source \n
action_and_metadata \n
optional_source \n
...

```
  - action může být index, create, update, delete
  - pozn. index vs. create
    - obojí zaindexuje dokument
    - rozdíl je v případě již existujícího dokumentu
    - index přepíše
    - create selže



# SPRÁVA DOKUMENTŮ

- dávkové zpracování
  - ukázka zaindexování dokumentů
  - pro každý dokument dvojice řádků
    - první akce a detaily
      - index, volitelně \_id
    - druhý samotný dokument
  - výsledky zobrazeny pro každý dokument

```
1 POST /_bulk
2 { "index": { "_index": "products", "_id": 200 } }
3 { "name": "Espresso Machine", "price": 199, "in_stock": 5 }
4 { "create": { "_index": "products", "_id": 201} }
5 { "name": "Milk Frother", "price": 149, "in_stock": 14 }
6
7
8
9 GET /products/_search
10 {
11   "query": {
12     | "match_all": {}
13   }
14 }
```

```
1 [
2   "took" : 138,
3   "errors" : false,
4   "items" : [
5     {
6       "index" : {
7         "_index": "products",
8         "_type" : "_doc",
9         "_id" : "200",
10        "_version" : 1,
11        "result" : "created",
12        "_shards" : {
13          "total" : 2,
14          "successful" : 2,
15          "failed" : 0
16        },
17        "_seq_no" : 25,
18        "_primary_term" : 1,
19        "status" : 201
20      }
21    },
22    {
23      "create" : {
24        "_index": "products",
25        "_type" : "_doc",
26        "_id" : "201",
27        "_version" : 1,
28        "result" : "created",
29        "_shards" : {
30          "total" : 2,
31          "successful" : 2,
32          "failed" : 0
33        },
34        "_seq_no" : 26,
35        "_primary_term" : 1,
36        "status" : 201
37      }
38    }
39  ]
40 }
```



# SPRÁVA DOKUMENTŮ

- dávkové zpracování
  - ukázka úpravy dokumentů
  - syntax jako u úpravy jednoho dokumentu
    - možnost skriptování

```
1 POST /_bulk
2 { "index": { "_index": "products", "_id": 200 } }
3 { "name": "Espresso Machine", "price": 199, "in_stock": 5 }
4 { "create": { "_index": "products", "_id": 201 } }
5 { "name": "Milk Frother", "price": 149, "in_stock": 14 }
6
7
8 POST /_bulk
9 { "update": { "_index": "products", "_id": 201 } }
10 { "doc": { "price": 129 } }
```

```
1 { "took" : 159,
2  "errors" : false,
3  "items" : [
4    {
5      "update" : {
6        "_index" : "products",
7        "_type" : "_doc",
8        "_id" : "201",
9        "_version" : 2,
10       "result" : "updated",
11       "shards" : {
12         "total" : 2,
13         "successful" : 2,
14         "failed" : 0
15       },
16       "seq_no" : 27,
17       "_primary_term" : 1,
18       "status" : 200
19     }
20   ]
21 }
22 }
```



# SPRÁVA DOKUMENTŮ

- dávkové zpracování
  - ukázka smazání dokumentů
  - neočekává druhý rádek

```
1 POST /_bulk
2 { "index": { "_index": "products", "_id": 200 } }
3 { "name": "Espresso Machine", "price": 199, "in_stock": 5 }
4 { "create": { "_index": "products", "_id": 201} }
5 { "name": "Milk Frother", "price": 149, "in_stock": 14 }

6
7
8 POST /_bulk
9 { "update": { "_index": "products", "_id": 201 } }
10 { "doc": { "price": 129 } }

11
12
13 POST /_bulk
14 { "delete": { "_index": "products", "_id": 200 } }| 
15
16
17
18 GET /products/_search
19 {
20 {
21 "query": {
22 | "match_all": {}
23 }
```

```
1 {
2   "took" : 181,
3   "errors" : false,
4   "items" : [
5     {
6       "delete" : {
7         "_index" : "products",
8         "_type" : "_doc",
9         "_id" : "200",
10        "_version" : 2,
11        "result" : "deleted",
12        "_shards" : {
13          "total" : 2,
14          "successful" : 2,
15          "failed" : 0
16        },
17        "_seq_no" : 28,
18        "_primary_term" : 1,
19        "status" : 200
20      }
21    ]
22  }
23 }
```



# SPRÁVA DOKUMENTŮ

- dávkové zpracování
  - všechny akce lze samozřejmě kombinovat
  - pokud jsou všechny akce prováděny nad stejným indexem, je možné specifikovat index v cestě

```
1 POST /_bulk
2 { "index": { "_index": "products", "_id": 200 } }
3 { "name": "Espresso Machine", "price": 199, "in_stock": 5 }
4 { "create": { "_index": "products", "_id": 200 }, }
5 { "name": "Milk Frother", "price": 149, "in_sto
6
7
8 POST /products/_bulk
9 { "update": { "_id": 201 } }
10 { "doc": { "price": 109 } }
11 { "delete": { "_id": 200 } }
12
13
14
15 GET /products/_search
16 {
17   "query": {
18     | "match_all": {}
19   }
20 }
```

Click to send request

```
1 { "took": 165,
2  "errors": false,
3  "items": [
4    {
5      "update": {
6        "_index": "products",
7        "_type": "_doc",
8        "_id": "201",
9        "_version": 3,
10       "result": "updated",
11       "shards": {
12         "total": 2,
13         "successful": 2,
14         "failed": 0
15       },
16       "seq_no": 32,
17       "primary_term": 1,
18       "status": 200
19     }
20   },
21   {
22     "delete": {
23       "_index": "products",
24       "_type": "_doc",
25       "_id": "200",
26       "_version": 4,
27       "result": "deleted",
28       "shards": {
29         "total": 2,
30         "successful": 2,
31         "failed": 0
32       },
33       "seq_no": 33,
34       "primary_term": 1,
35       "status": 200
36     }
37   }
38 ]
39 }
```





# SPRÁVA DOKUMENTŮ

- dávkové zpracování
  - pozor na správný HTTP Content-Type
    - Content-Type: application/x-ndjson
  - každý řádek musí končit znakem newline
    - pozor na poslední řádek
    - \n nebo \r\n
  - selhaná akce neovlivní ostatní
    - nedojde ke zrušení dávkového zpracování
    - nejedná se o atomickou operaci
- použití
  - velké množství zápisů ve stejný čas
    - import dat, modifikace velkého množství dat
  - efektivnější než jednotlivé dotazy
  - skript generující žádosti



# IMPORT DOKUMENTŮ ZE SOUBORU

- s pomocí cURL a dávkového zpracování

```
curl -u elastic -H "Content-Type: application/x-ndjson" -XPOST  
https://localhost:9200/products/_bulk --data-binary  
"@products-bulk.json"
```

po opravení chyby bez ~~k~~  
a s certifikátem

- s certifikátem

```
1 {"index":{"_id":1}}  
2 {"name":"Wine - Maipo Valle Cabernet","price":152,"in_stock":38,"sold":47,"tags":[]}  
"Alcohol","Wine"],"description":"Aliquam augue quam, sollicitudin vitae,  
consectetur eget, rutrum at, lorem. Integer tincidunt ante vel ipsum. Praesent  
blandit lacinia erat. Vestibulum sed magna at nunc commodo placerat. Praesent  
blandit. Nam nulla. Integer pede justo, lacinia eget, tincidunt eget, tempus vel,  
pede. Morbi porttitor lorem id ligula.", "is_active":true,"created":"2004/05/13"}  
3 {"index":{"_id":2}}  
4 {"name":"Tart Shells - Savory","price":99,"in_stock":10,"sold":430,"tags":[]},  
"description":"Pellentesque at nulla. Suspendisse potenti. Cras in purus eu magna  
vulputate luctus. Cum sociis natoque penatibus et magnis dis parturient montes,  
nascetur ridiculus mus. Vivamus vestibulum sagittis sapien. Cum sociis natoque  
penatibus et magnis dis parturient montes, nascetur ridiculus mus. Etiam vel augue.  
Vestibulum rutrum rutrum neque. Aenean auctor gravida sem.", "is_active":true,  
"created":"2007/10/14"}  
5 {"index":{"_id":3}}  
6 {"name":"Kirsch - Schloss","price":25,"in_stock":24,"sold":215,"tags":[]},  
"description":"In eleifend quam a odio.", "is_active":true,"created":"2000/11/17"}
```

```
D:\Prezentace\_ostatní\DPB\DPB_stuff>curl -k -u elastic -H "Content-Type: application/x-ndjson" -XPOST https://localhost:9200/products/_bulk --data-binary "@products-bulk.json"  
Enter host password for user 'elastic':  
{"took":1035,"errors":false,"items":[{"index":{"_index":"products","_id":"1","_version":1,"result":"created","_shards":{"total":2,"successful":1,"failed":0},"_seq_no":0,"_primary_term":1,"status":201}},{"index":{"_index":"products","_id":"2","_version":1,"result":"created","_shards":{"total":2,"successful":1,"failed":0},"_seq_no":1,"_primary_term":1,"status":201}},{"index":{"_index":"products","_id":"3","_version":1,"result":"created","_shards":{"total":2,"successful":1,"failed":0},"_seq_no":2,"_primary_term":1,"status":201}]},"in
```



# ÚVOD DO PROHLEDÁVÁNÍ

- dva způsoby psaní vyhledávacích dotazů
  - Request UI
    - vyhledávací dotaz jako parametr
    - omezenější funkcionality
      - ale i tak zvládá složité dotazy
    - méně přehledné
      - obzvlášť u složitých dotazů
  - rychlé testování během vývoje
    - dotazy z příkazové řádky

```
1 GET /products/_search?q=* ▶ ⌂
```

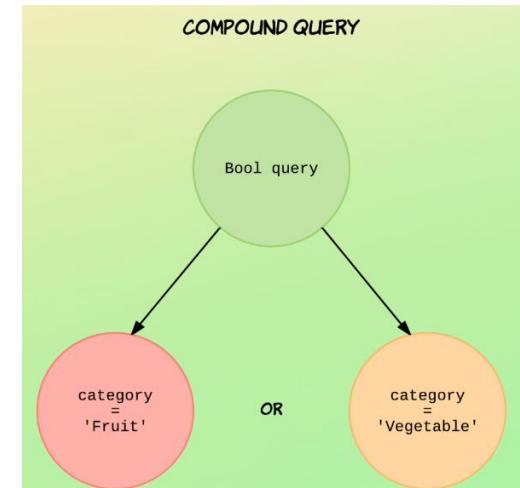
```
1 GET /products/_search?q=name:Lobster ▶ ⌂
```

```
1 GET /products/_search?q=tags:Meat AND name:Tuna ▶ ⌂
```



# ÚVOD DO PROHLEDÁVÁNÍ

- dva způsoby psaní vyhledávácích dotazů
  - Query DSL
    - předání dotazovacího JSON objektu
    - hlavní způsob dotazování
    - dvě hlavní skupiny dotazů
    - leaf queries
      - vyhledávají hodnotu v konkrétních polích
      - např. kategorie Ovoce
    - compound queries
      - skládají se z leaf queries nebo dalších compound queries
      - rekuzivní
      - např. kategorie Ovoce nebo (boolean) Zelenina



<https://www.udemy.com/course/elasticsearch-complete-guide/>

```
1 GET /products/_search ▶ 🔍  
2 {  
3   "query": {  
4     "match_all": {}  
5   }  
6 }
```

```
1 GET /products/_search ▶ 🔍  
2 {  
3   "query": {  
4     "match": {  
5       "description": "red wine"  
6     }  
7   }  
8 }
```



# ÚVOD DO PROHLEDÁVÁNÍ

- reprezentace výsledků
  - took
    - čas provedení dotazu [ms]
  - timed\_out
    - boolean vypršení času na dotaz
  - \_shards
  - hits
    - hlavní výsledek hledání
    - total
      - počet dokumentů odpovídajících vyhledávání
    - max\_score
      - maximální skóre relevance
    - hits
      - pole samotných výsledků
      - \_score
        - skóre relevance
      - vyjadřuje míru, s jakou výsledek odpovídá dotazu

```
1  {
2    "took" : 1,
3    "timed_out" : false,
4    "_shards" : {
5      "total" : 1,
6      "successful" : 1,
7      "skipped" : 0,
8      "failed" : 0
9    },
10   "hits" : {
11     "total" : {
12       "value" : 1,
13       "relation" : "eq"
14     },
15     "max_score" : 8.577639,
16     "hits" : [
17       {
18         "_index" : "products",
19         "_type" : "_doc",
20         "_id" : "65",
21         "_score" : 8.577639,
22         "_source" : {
23           "name" : "Tuna - Bluefin",
24           "price" : 27,
25           "in_stock" : 26,
26           "sold" : 378,
27           "tags" : [
28             "Meat"
29           ],
30           "description" : "Integer pede justo, lacinia
31             eget, tincidunt eget, tempus vel, pede.
32             Morbi porttitor lorem id ligula.
33             Suspendisse ornare consequat lectus.",
34           "is_active" : false,
35           "created" : "2015/03/23"
36         }
37     }
38   }
```



# ÚVOD DO PROHLEDÁVÁNÍ

- skóre relevance (\_score)
  - vyjadřuje, jakou měrou výsledek vyhledávání odpovídá dotazu
  - jsou podle něj řazeny výsledky vyhledávání
    - cílem není jen vrátit výsledky vyhledávání, ale vrátit **relevantní** výsledky
    - rozdíl oproti klasickým databázovým systémům
- postup v Elasticsearch
  - boolean model
  - nejprve jsou nalezeny všechny dokumenty, které odpovídají dotazu
    - vyřadí ostatní dokumenty z počítání skóre relevance
  - pro vybrané dokumenty je následně počítáno skóre relevance



<https://www.udemy.com/course/elasticsearch-complete-guide/>





# ÚVOD DO PROHLEDÁVÁNÍ

- jak je skóre relevance počítáno?
  - liší se podle typu vyhledávacího dotazu
  - možnost výběru algoritmu
    - i možná úprava skórování
  - dosud využíván algoritmus TF/IDF
    - Term Frequency / Inverse Document Frequency
  - v současnosti algoritmus Okapi BM25
  - oba algoritmy jsou si velmi podobné
    - založené na stejném principu
    - Okapi BM25 řeší určitě nedostatky TF/IDF



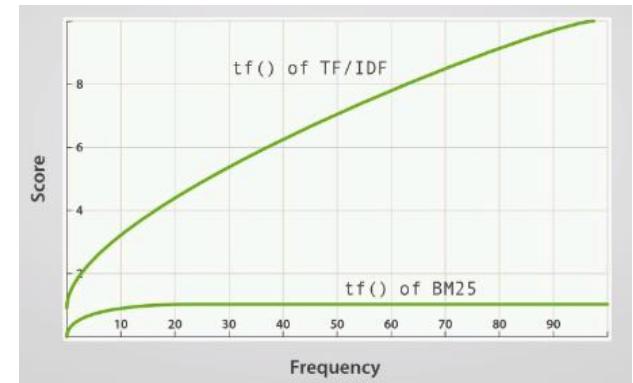
# ÚVOD DO PROHLEDÁVÁNÍ

- algoritmus Term Frequency / Inverse Document Frequency
  - Term Frequency
    - vyjadřuje, kolikrát se hledaný výraz nachází v daném poli dokumentu
    - vyšší výskyt znamená vyšší relevanci
    - Pes vrtí ocasem. Pes štěká. Pes kouše.
      - pes – 3×, vše ostatní – 1×
  - Inverse Document Frequency
    - vyjadřuje, jak často se hledaný výraz nachází v indexu
      - přes všechny dokumenty
    - vyšší výskyt znamená nižší relevanci
      - častá slova jsou méně důležitá, např. a, já, je, se, ...
  - norma délky pole (Field-Length Norm)
    - vyjadřuje, jak dlouhé pole je
    - kratší pole znamená vyšší relevanci
      - výraz v 5 slovném textu vs. výraz v 1000 slovech
  - TF, IDF, FLN počítány a uloženy při indexaci (přidání, update dokumentu)



# ÚVOD DO PROHLEDÁVÁNÍ

- vylepšení v Okapi BM25
  - lepší práce se stopslovy (stop words)
    - častá slova, která nesou jen minimum informací o relevantnosti dokumentu k vyhledávání
    - dříve kompletně vyneschávána
    - i tak ale nesou určitou informaci
    - př. dokument s polem obsahujícím kompletní text knihy
      - velké množství stopslov
      - u TF/IDF by mohlo dojít k převážení stopslov
      - BM25 řeší pomocí Nonlinear Term Frequency Saturation
    - Nonlinear Term Frequency Saturation
      - horní limit na převážení podle počtu výskytů
      - např. váha pro 30 výskytů vs. 1000 výskytů je téměř stejná
      - umožňuje ponechání stopslov
  - vylepšená norma délky slova
    - bere v potaz průměrnou délku pole
  - laditelné parametry



<https://www.udemy.com/course/elasticsearch-complete-guide/>



# ÚVOD DO PROHLEDÁVÁNÍ

- debuggování neočekávaných výsledků vyhledávání
  - neočekávané výsledky mohou být častým problémem
  - řešením může být \_explain API
    - na základě \_id dokumentu a dotazu zobrazí proč byl nebo nebyl dokument nalezen

```
1 GET /products/_explain/1 ▶ 🔍
2 {
3   "query": {
4     "match": {
5       "name": "lobster"
6     }
7   }
8 }
```

```
1 { "_index": "products",
2   "_type": "_doc",
3   "_id": "1",
4   "matched": false,
5   "explanation": {
6     "value": 0.0,
7     "description": "no matching term",
8     "details": []
9   }
10 }
11 }
```



# ÚVOD DO PROHLEDÁVÁNÍ

- kontext dotazů
  - dotaz může být položen ve dvou různých kontextech
  - dotazovací kontext (query context)
    - ptáme se: „Jak moc dokumenty odpovídají dotazu?“
    - nejprve je zjištěno, jestli dokument odpovídá dotazu
    - následně je vyhodnoceno skóre relevance
    - články věnované vyhledávanému termínu, ...
  - filtrovací kontext (filter context)
    - ptáme se: „Vyhovují dokumenty dotazu?“
    - binární – ano / ne
    - odpovídá prvnímu kroku z dotazovacího kontextu
    - není počítáno skóre relevance
    - mohou být cachované
    - datumy, statusy, rozsahy, ...



# ÚVOD DO PROHLEDÁVÁNÍ

- dotazovací typy
  - dotazy na úrovni termínů (term level queries)

```
1 GET /products/_search ▶ 🔍  
2 {  
3   "query": {  
4     "term": {  
5       "name": "lobster"  
6     }  
7   }  
8 }
```

```
10  "hits" : {  
11    "total" : {  
12      "value" : 5,  
13      "relation" : "eq"  
14    },  
15    "max_score" : 6.035804,  
16    "hits" : [  
17      {  
18        "_index" : "products",  
19        "_type" : "_doc",  
20        "_id" : "19",  
21        "_score" : 6.035804,  
22        "_source" : {  
23          "name" : "Lobster - Live",
```

```
1 GET /products/_search ▶ 🔍  
2 {  
3   "query": {  
4     "term": {  
5       "name": "Lobster"  
6     }  
7   }  
8 }
```

```
1 {  
2   "took" : 1,  
3   "timed_out" : false,  
4   "_shards" : {  
5     "total" : 1,  
6     "successful" : 1,  
7     "skipped" : 0,  
8     "failed" : 0  
9   },  
10  "hits" : {  
11    "total" : {  
12      "value" : 0,  
13      "relation" : "eq"  
14    },  
15    "max_score" : null,  
16    "hits" : [ ]  
17  }  
18 }
```

- fulltextové dotazy (full-text queries)

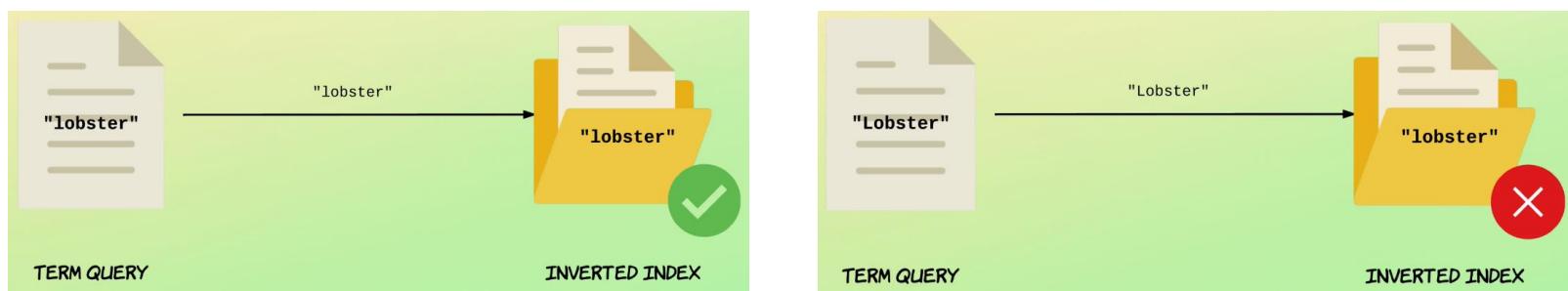
```
1 GET /products/_search ▶ 🔍  
2 {  
3   "query": {  
4     "match": {  
5       "name": "Lobster"  
6     }  
7   }  
8 }
```

```
10  "hits" : {  
11    "total" : {  
12      "value" : 5,  
13      "relation" : "eq"  
14    },  
15    "max_score" : 6.035804,  
16    "hits" : [  
17      {  
18        "_index" : "products",  
19        "_type" : "_doc",  
20        "_id" : "19",  
21        "_score" : 6.035804,  
22        "_source" : {  
23          "name" : "Lobster - Live",
```



# ÚVOD DO PROHLEDÁVÁNÍ

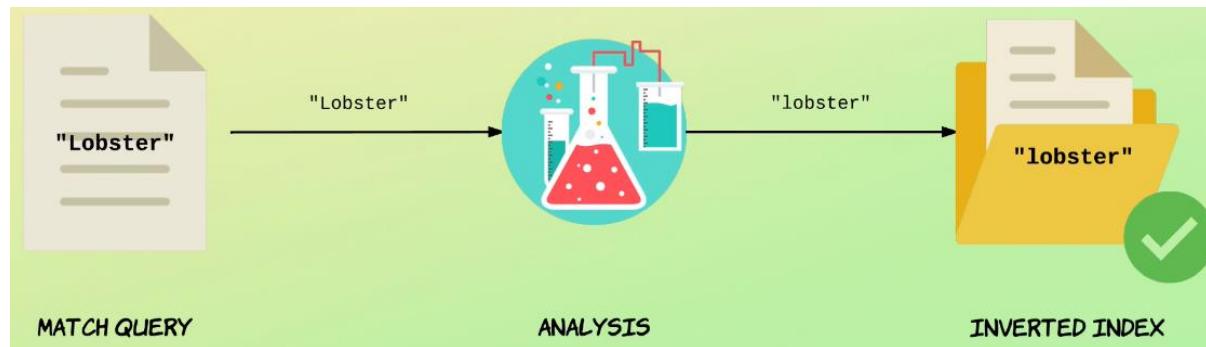
- dotazovací typy
  - dotazy na úrovni termínů (term level queries)
    - vyhledávají přesnou hodnotu vůči invertovanému indexu (po analýze)
      - ne vůči přímo cílovému dokumentu
    - vyhledávaná hodnota není analyzována
    - vhodné např. pro datumy, čísla, ...
      - ne pro řetězce



<https://www.udemy.com/course/elasticsearch-complete-guide/>

# ÚVOD DO PROHLEDÁVÁNÍ

- dotazovací typy
  - fulltextové dotazy (full-text queries)
    - vyhledávaná hodnota je analyzována stejným analyzárem jako invertovaný index
    - je možné najít jen hodnoty v invertovaném indexu
    - vhodné pro fulltextové vyhledávání
      - řetězce



<https://www.udemy.com/course/elasticsearch-complete-guide/>



# A PŘÍŠTĚ?

- Elasticsearch
  - invertované indexy
  - analýza a mapování
  - prohledávání





Děkuji za pozornost.  
Otázky?

