

Západočeská univerzita v Plzni
Fakulta aplikovaných věd
Katedra informatiky a výpočetní techniky

Bakalářská práce

Vizualizace dat z nástroje ElasticSearch

ZÁPADOČESKÁ UNIVERZITA V PLZNI

Fakulta aplikovaných věd

Akademický rok: 2019/2020

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: **Ondřej ANDĚL**
Osobní číslo: **A17B0177P**
Studijní program: **B3902 Inženýrská informatika**
Studijní obor: **Informatika**
Téma práce: **Vizualizace dat z nástroje Elasticsearch**
Zadávací katedra: **Katedra informatiky a výpočetní techniky**

Zásady pro vypracování

1. Prostudujte framework Elasticsearch a vizualizační nástroje, které lze s Elasticsearch použít.
2. Prozkoumejte dodaná data a srovnajte vybrané vizualizační nástroje z hlediska jejich využití k zobrazení dodaných dat.
3. Využijte vybraný nástroj k vytvoření vizualizace. Zaměřte se na možnosti integrace do webových aplikací a interakce s grafy.
4. Ověřte dosažené řešení na netriviální množině dat a kriticky zhodnoťte dosažené výsledky.

Rozsah bakalářské práce: **doporuč. 30 s. původního textu**
Rozsah grafických prací: **dle potřeby**
Forma zpracování bakalářské práce: **tištěná**

Seznam doporučené literatury:

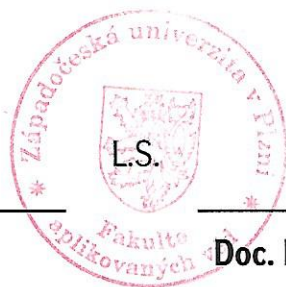
Dodá vedoucí bakalářské práce.

Vedoucí bakalářské práce: **Ing. Miloslav Konopík, Ph.D.**
Katedra informatiky a výpočetní techniky

Datum zadání bakalářské práce: **7. října 2019**
Termín odevzdání bakalářské práce: **7. května 2020**



Doc. Dr. Ing. Vlasta Radová
děkanka



Doc. Ing. Přemysl Brada, MSc., Ph.D.
vedoucí katedry

Prohlášení

Prohlašuji, že jsem bakalářskou práci vypracoval samostatně a výhradně s použitím citovaných pramenů.

V Plzni dne 7. května 2020

Ondřej Anděl

Abstract

ElasticSearch is a full-text search engine, but as a app itself, it lacks the means to visualize data in a comprehensible format. The goal of this study is to research and describe other software which enables visualization of the data stored in ElasticSearch. It was established that the compared apps should not send data to foreign server. All visualization should have the option to filter in them (they should be interactive) and enable exporting of said visualization (preferably in standalone website). While the technology is not required to be free it is preferred so.

Abstrakt

ElasticSearch je fulltextový vyhledávač, který ovšem postrádá nástroje pro vizualizaci dat a tudíž i jejich zobrazení v prezentovatelné formě. Cílem této práce je vyhledat, prostudovat a popsat jiné aplikace umožňující vizualizovat data uložená v ElasticSearch. Bylo domluveno, že porovnávané aplikace nemají posílat data na vnější server. Všechny vizualizace mají umožňovat filtrování (mají být interaktivní) a zároveň mají umožnit export zmíněných vizualizací (ideálně do samostatné webové stránky). Vybraná technologie nemusí podléhat volným licencím, ale je to preferované.

Poděkování

Rád bych poděkoval Ing. Miloslavu Konopíkovi, Ph.D. za cenné rady, věcné připomínky a vstřícnost při konzultacích a vypracování bakalářské práce.

Obsah

| | | |
|----------|---|-----------|
| 1 | Úvod | 8 |
| 2 | Elastic Stack | 9 |
| 2.1 | ElasticSearch | 10 |
| 3 | Vizualizační nástroje | 13 |
| 3.1 | Kibana | 13 |
| 3.1.1 | Agregace | 13 |
| 3.1.2 | Vizualizační metody | 15 |
| 3.1.3 | Vizualizační skupiny | 20 |
| 3.2 | Grafana | 24 |
| 3.2.1 | Systém vizualizací | 25 |
| 3.3 | Knowi | 27 |
| 3.4 | Jazyk R | 29 |
| 3.4.1 | RStudio | 31 |
| 3.5 | Plotly | 32 |
| 3.5.1 | Falcon | 32 |
| 3.5.2 | Prostředí Plotly | 33 |
| 3.6 | Aplikace nevyhovující zadání | 35 |
| 4 | Výsledky porovnání | 36 |
| 5 | Testovací data | 38 |
| 5.1 | Připojení Grafany k ElasticSearch | 39 |
| 5.2 | Vlastní vizualizace | 41 |
| 5.3 | Detekované nedostatky | 43 |
| 6 | Závěr | 44 |
| | Literatura | 45 |

1 Úvod

V dnešní době, kdy jsou jednotlivé firmy schopné vyprodukovat obrovské množství dat denně, je velmi důležité co nejlépe usnadnit vyhledávání v těchto datech. Na trhu lze najít mnoho aplikací, poskytující možnost hledání nad databázemi dokumentů, mnoho z nich ale omezuje velikost přijímaných dat, velikost úložiště, počet uživatelů nebo nabízejí pouze komerční licence. Díky tomu se během posledních let prosadil nástroj ElasticSearch, který nabízí základní nástroje k fulltextovému hledání již pod volnou licenci.

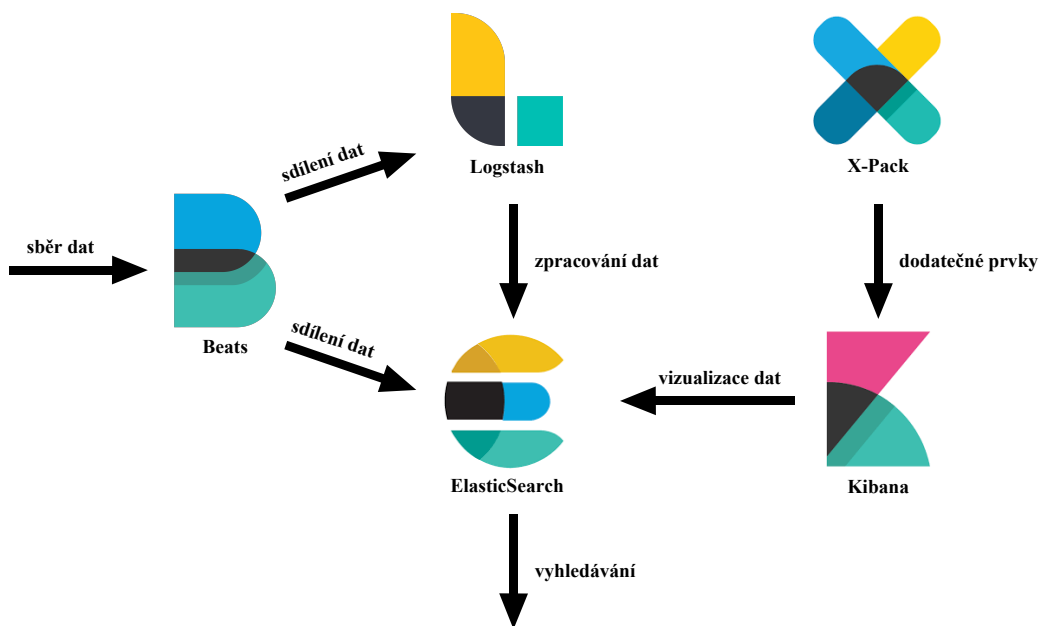
Mnoho firem požaduje nicméně nejen funkční vyhledávání nad daty, ale i automatické přidávání dat, uživatelsky přívětivé prostředí a pochopitelné zobrazení dat. Tyto funkcionality ElasticSearch sám nenabízí, ale díky velké modularitě je možné ElasticSearch propojit s technologiemi, které jsou schopné velmi efektivně přidávat data do databáze (například Logstash), nebo dříve zmíněná data zobrazit a usnadnit jejich ovládání (Kibana). Nástroje testované naší studií musí automaticky provádět dotazy nad nástrojem ElasticSearch, podporovat interaktivní filtrování dat a umožnit integrování do externích stránek.

Přestože je možné ElasticSearch používat i bez těchto technologií, přidání vizualizačních nástrojů umožní rychlejší a názornější analýzu dat. Vizualizační nástroje navíc nabízejí i grafické rozhraní, díky čemuž mohou aplikaci obsluhovat i uživatelé neznalí vnitřní struktury dat ElasticSearch. Vizualizačních nástrojů, jež jsou schopny spolupracovat s ElasticSearch, je velké množství a jejich kvalita se liší, z toho důvodu se pokusím vybrat a kriticky ohodnotit nejlepší z nich.

2 Elastic Stack

Elastic Stack je název pro skupinu serverových aplikací spadajících pod Elasticsearch. Některé z nich jsou schopny pracovat samostatně, jiné vyžadují ke správné funkcionalitě některé ze zbylých komponent Elastic Stacku. Základní technologie a prvky Elastic Stacku jsou dostupné zdarma, ale k zpřístupnění plné funkcionality je nutné koupit si placenou licenci.

Některé firmy a programátoři stále využívají název ELK Stack, který ale označuje dřívější verzi tvořenou pouze třemi nástroji aktuálního Elastic Stacku. Tyto nástroje jsou Elasticsearch, Logstash a Kibana. Elastic Stack navíc nabízí nástroj Beats. Někdy se uvádí, že součástí Elastic Stack je i nástroj zvaný X-Pack, ale zde se jedná pouze o další rozšíření nástroje Kibana a dnes se již X-Pack většinou nerozlišuje jako samostatná část. Vazby mezi spolupracujícími komponentami Elastic Stacku je možné vidět na obrázku 2.1.[2]



Obrázek 2.1: Spolupráce mezi komponentami Elastic Stacku

2.1 Elasticsearch

ElasticSearch je nástroj pro fulltextové vyhledávání, to znamená, že vyhledává v databázích nebo celých textových souborech. Jedná se o technologii využívající RESTful rozhraní, tento nástroj tudíž umožňuje jednoduchou manipulaci s daty na serveru a vyhledávání v nich. REST (Representational state transfer) je označení pro datově orientovanou distribuovanou architekturu sloužící k obsluze serveru pomocí jednoduchých HTTP volání. Při využívání aplikace závislé na této architektuře je její stav a chování popsáno v klíčové abstrakci (resource). Každý resource má vlastní unikátní identifikátor, často URL. S klíčovou abstrakcí je manipulováno pomocí čtyř operací Create, Read, Update a Delete.[16] V ElasticSearch je tento resource uložen ve formátu JSON.

| | | Termíny | | | | | | | | | |
|-----------|---|---------|-------|------|------|-------|------|----|----|--|--|
| | | kamarád | modré | nové | kolo | druhé | také | to | je | má | |
| Dokumenty | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | Kamarád má nové modré kolo. | |
| | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | To modré kolo je nové. | |
| | 0 | 1 | 1 | 2 | 1 | 1 | 2 | 1 | 0 | To nové kolo je modré, to druhé kolo také. | |
| | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | Kamarád má také druhé kolo. | |

Obrázek 2.2: Příklad metody invertovaného indexu

Data ElasticSearch nejsou ukládána formou řádků a sloupců, ale jako serializovaná data formátu JSON. Každý dokument je ukládán jako množina polí a jednotlivé množiny dokumentů se nazývají indexy. K maximálnímu urychlení requestů používá ElasticSearch pro jednotlivé datové typy různé způsoby ukládání. Pro textové řetězce je použita metoda invertovaných indexů. To znamená, že jsou jednotlivé prvky dokumentů rozděleny na krátké úseky, takzvané termíny, a je k nim přiřazeno id dokumentů, v nichž jsou daná slova přítomná. Index tvořený množinou dokumentů je tedy reprezentován maticí, kde sloupce představují jednotlivé termíny a řádky jednotlivé dokumenty. Jak je patrné z obrázku 2.2, matice obsahuje v buňkách četnosti výskytů termínů v každém z dokumentů. Všechny termíny jsou ukládány pouze v malých písmenech. Naopak číselné hodnoty a data reprezentující polohu (tzv. geo data) jsou ukládány v BKD stromech. Tento mechanismus zdatelně zrychluje jednotlivá vyhledávání a analýzy, protože nemusí

být prohledáván každý dokument zvlášť. Zároveň je nad takto upravenými dokumenty jednoduché aplikovat jednotlivé filtry.[11]

Existují ale případy, kdy není vhodné rozdělovat textová data do termínů, například víceslovné názvy měst (podle mezer) nebo ip adresy (podle teček). Z tohoto důvodu existuje v Elasticsearch možnost využít původní tvar daného pole. Ve starších verzích Elasticsearch (starší než verze 5) byly obě možnosti dohledatelné pod názvem řetězec (string) a rozlišovali se dodatečným popiskem analyzed (pro data rozdělená do termínů) a not_analyzed (pro původní nerozparovaný řetězec). V novějších verzích se používá označení text (pro rozdělené prvky) a keyword (původní znění).

ElasticSearch nabízí jednoduché stavební bloky, takzvané agregace. Tyto agregace slouží k získání informací o množině dokumentů. Aplikace nabízí čtyři různé typy agregací.

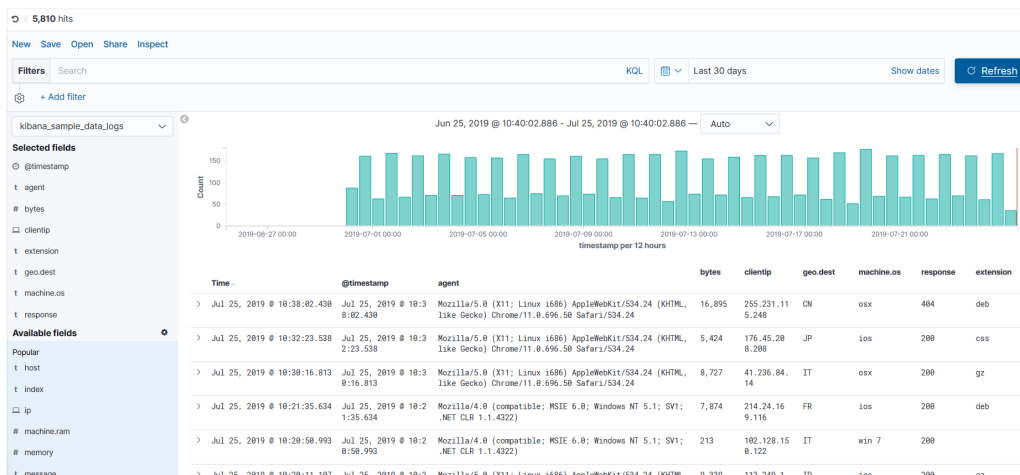
Prvním agregačním typem je Bucketing, jež rozděluje dokumenty do skupin, takzvaných bucketů. Každý bucket je určen speciálním klíčem a podmínkou, při rozdělování dokumentů jsou testovány všechny přítomné buckety. Je-li podmínka splněna, dokument patří do dané skupiny. Na konci agregačního procesu skončíme se seznamem bucketů, jež mají pod sebou sady dokumentů.

Druhým agregačním typem je Metric, jež se stará o spočtení metrik nad sadou dokumentů nebo termínů. Jinak řečeno, tento agregační typ přiřazuje skupině dokumentů (či termínů) číselnou hodnotu reprezentující její vlastnosti.

Třetím typem je Matrix. Tento typ zpracovává hodnoty několika polí do maticových výsledků. Poslední typ agregací se nazývá Pipeline a pracuje přímo nad ostatními agregacemi, z nichž zpracovává informace do výstupního stromu. Typ zpracovávané informace se liší podle specifického typu Pipeline agregace.

ElasticSearch podporuje několik typů vyhledávání (strukturované, nestrukturované, atd.) a umožňuje jejich kombinaci. Tím je pro uživatele snadnější najít přesně ty data, jež jsou pro ně relevantní. Z důvodu prevence poškození a ztráty dat, je možné si nastavit ukládání dat do dvou nezávislých clusterů. Při poruše jednoho clusteru je druhý automaticky použit pro obnovení dat.

ElasticSearch je schopný vyhledávat nejen nad jedním zařízením, ale i nad množinou serverů. Veškerá data jsou automaticky filtrována podle času přidání do indexu (v Kibaně je příslušný ovládací prvek umístěn v pravém horním rohu aplikace). Všechny nalezené výsledky (označované jako hits) jsou zobrazeny v původní neupravené formě (JSON), použitím externích nástrojů (např. Kibana) lze ale zobrazit pouze relevantní pole databáze



Obrázek 2.3: Uložené vyhledávání v Kibaně s roztříděnými daty

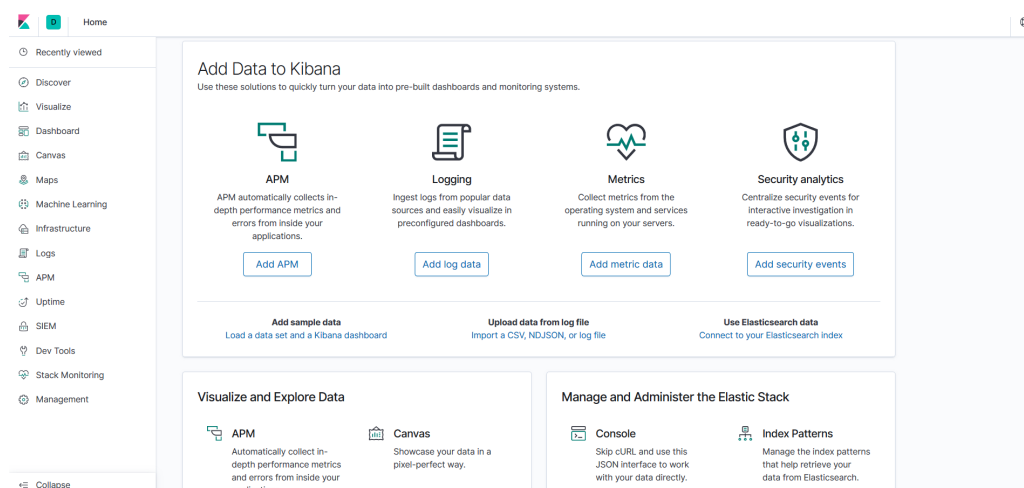
v uživatelsky přívětivé formě (formát řádkových a sloupcových dat). Takto upravené vyhledání je možné si uložit a kdykoliv později zobrazit nebo exportovat do formátů JSON nebo CSV. Data v uloženém vyhledávání jsou automaticky upravována, je-li do clusteru přidán dokument, dokument bude přítomen i v příslušném vyhledávání. Příklad uloženého vyhledávání nad vzorovými daty je vidět na obrázku 2.3. Pro větší názornost byl v daném obrázku použit navíc nástroj Kibana poskytující lepší uživatelské rozhraní.

Přestože je aplikace vyvinutá z Apache Lucene a tudíž napsaná v Javě, ovládat ji lze nejenom za využití tohoto programovacího jazyka, ale lze využít i jiných programovacích a skriptovacích jazyků. Oblíbenými prostředky jsou zejména dotazovací jazyk SQL a DSL. Oba z těchto prostředků jsou podporovány díky svojí velké rozšířenosti v obvyklých databázových systémech. SQL dotazy je možné přetransformovat aplikací do DSL dotazů. Aplikací podporované SQL je často nazýváno Elasticsearch SQL, neboť na rozdíl od původní verze SQL nepodporuje zápis a úpravu dat. Elasticsearch query slouží pouze pro čtení dat. Mimo to lze ovšem využít i nástroje Kibana, který ovládání Elasticsearch zjednodušuje a přibližuje uživatelům. Základními vyhledávacími jazyky aplikace nicméně stále zůstávají Lucene query a jeho zjednodušená verze Kibana query.[4]

3 Vizualizační nástroje

3.1 Kibana

Formát dat v aplikaci Elasticsearch může být pro většinu uživatelů nepřehledný a s větším počtem dat může být téměř nemožné se v nich orientovat. Z tohoto důvodu byl autory Elastic Stacku vytvořen nástroj Kibana, který je určen k vizualizacím dat z Elasticsearch. Kibana nabízí grafické uživatelské rozhraní, tudíž umožňuje mnohem jednodušší ovládání zbylých nástrojů Elastic Stacku. Úvodní okno, na kterém jsou nabízeny nejčastěji užívané nástroje je k vidění na obrázku 3.1.



Obrázek 3.1: Úvodní okno Kibany

Kibana také nabízí možnost detekce anomálií v datech nebo jejich specifikacích, díky tomu lze minimalizovat tyto odchylky a maximalizovat profit a efektivitu firmy. K analýze a další detekci těchto odchylek je používáno strojové učení. Tato technologie je dokonce schopná předpovídat další přírůstky dat a jejich charakter, což umožňuje snadnější přípravu na krizové situace.[7]

3.1.1 Agregace

Vizualizace nástroje Kibana vyžadují ke svému správnému vykreslení přiřazení agregace. Agregace Kibany rozdělujeme do dvou skupin, většina grafů vyžaduje přiřazení agregace typu Bucket na osu x a agregace typu Metric

na osu y. Každá z těchto skupin obsahuje různé agregace, kterým se budu věnovat níže, nejprve je ale nutné představit rozdíl mezi těmito dvěma typy.

Bucket agregace jsou určeny k rozdělování dokumentů načtených do Elasticsearch do menších skupin, jež jsou nazývány buckety. Jednotlivé buckety se mohou lišit počtem obsažených dokumentů a jejich vlastnostmi. Každý dokument může být obsažen v několika různých bucketech.

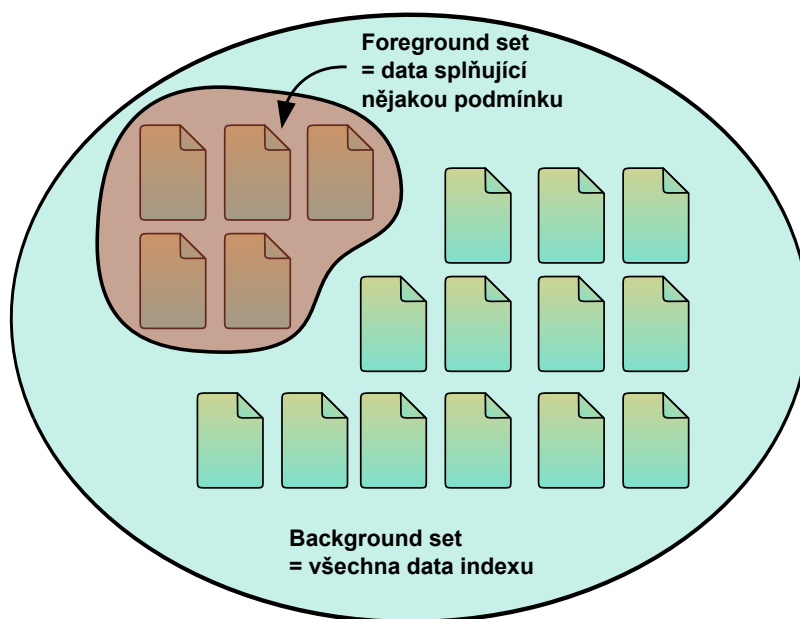
Metric agregace naopak buckety nevytváří a předpokládají již jejich existenci. Hlavním úkolem těchto agregací je přiřadit jednotlivým skupinám číselnou hodnotu, která nějakým způsobem reprezentuje počet nalezených výsledků v bucketech.

ElasticSearch a Kibana momentálně nabízí devět Bucket agregací a jedenáct hlavních Metric agregací. Jak již bylo podotknuto výše, Metric agregace předpokládají existenci skupin, proto nejprve představím agregace typu Bucket.

Agregace Histogram se zabývá pouze daty ve formátu čísel a je schopna rozdělit dokumenty do skupin podle uživatelem zadaného nebo automatického intervalu. Podobným způsobem funguje i agregace Date histogram, která rozděluje dokumenty podle časového intervalu. Kibana dále nabízí agregace, ve kterých je možné určit přímo rozsah jednotlivých skupin, tyto agregace jsou dále rozlišeny podle typu dat a jsou to Range (pro čísla), Date Range (pro datum), IPv4 Range (pro ip adresy). Agregace Geohash je dostupná pouze pro Coordinate Map grafy a rozděluje data podle geografické polohy. Filters agregace rozdělí dokumenty podle uživatelem zadaných Lucene filtrů. Uživatelé mohou také rozdělit data do bucketů podle obsažených slov nebo sekvencí znaků, k tomu jsou využívány poslední dvě agregace Terms a Significant terms. Tyto dvě agregace se liší tím, že Terms rozdělí data přímo podle řetězců, zatímco Significant terms statisticky porovná frekvenci výskytu vybraných dat (foreground set) vůči frekvenci výskytu dat v celku (background set). Díky tomu se Significant terms hodí ke zkoumání anomálií dat. Pro snazší pochopení této poslední agregace je koncept nastíněn obrázkem 3.2.

Nejjednodušší z Metric agregací je Count, který pouze spočte počet dokumentů v každém bucketu. Další agregací je Unique count, jež slouží k spočítání rozdílných hodnot v poli. Agregace Sum, jak již název napovídá, sečte všechny hodnoty daného pole pro každý bucket. Agregace Average vypočte průměrnou hodnotu z daného pole a agregace Median zase vypočte medián z téhož pole. Pro výběr extrémních hodnot Kibana nabízí agregace Min a Max, které fungují i nad daty reprezentující čas.

Agregace Standart Deviation na rozdíl od dříve zmíněných typů přiřazuje skupinám dvě čísla reprezentující horní a dolní hranici odchylky. Také agre-



Obrázek 3.2: Significant terms agregace

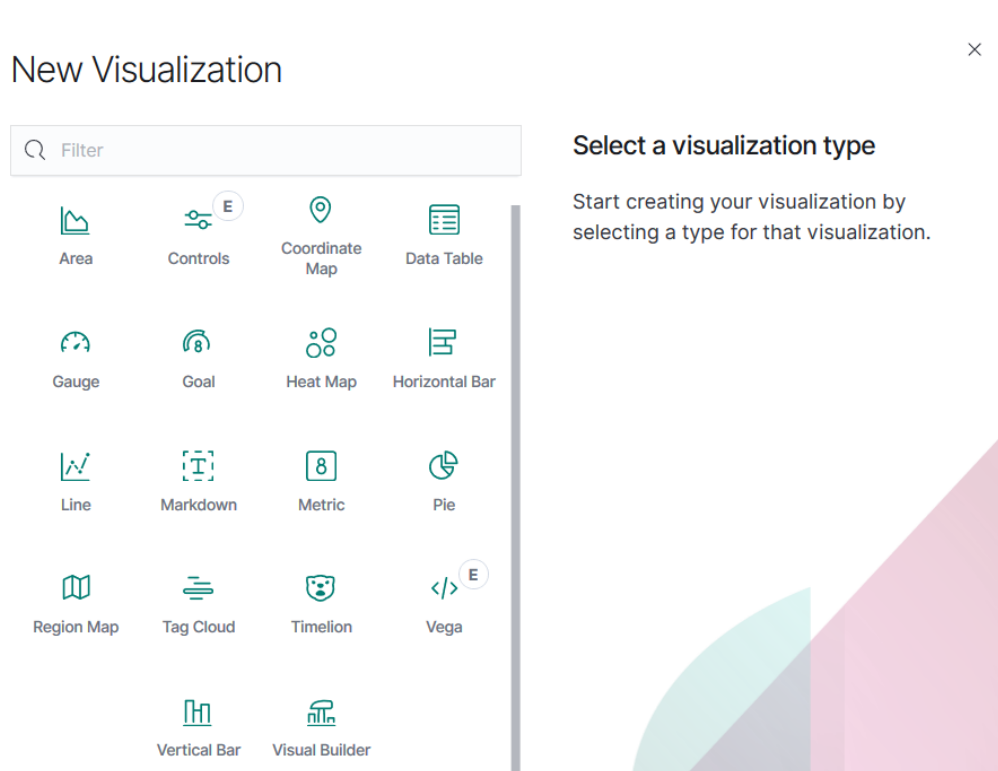
gace Percentiles vrací více hodnot, každá z hodnot reprezentuje počet dokumentů, pro něž je hodnota menší nebo rovna zadaným procentům. Opakem této skupiny je Percentil rank, který přiřadí bucketům číselnou hodnotu reprezentující, kolik procent z indexu daný bucket představuje. Poslední agregací je Top hit agregace se používá jako sekundární modifikátor k rozřídění dat.[8]

3.1.2 Vizualizační metody

Nástroj Kibana nabízí v momentální verzi osmnáct vizualizačních elementů. K výběru je například z histogramů, koláčových grafů, heatmap, ale také lze data vizualizovat podle místa nebo času uploadu, v mapě nebo časové síti. Pokud nabízené grafy nestačí uživatelským potřebám, je možné si za využití Vega gramatiky vytvořit vlastní vizualizace.[8]

Vega gramatika je speciální deklarativní jazyk sloužící k vytváření a sdílení interaktivní vizualizací dat. Tento jazyk popisuje vizualizační prvky prostřednictvím JSON formátu, které jsou posléze vykresleny prostřednictvím JavaScriptu. Prostřednictvím Vega je možné vytvořit statické i interaktivní vizualizace.[18]

Uživatelé bez předchozích zkušeností s Kibanou si mohou prohlédnout nabídku hotových vizualizačních grafů nad testovacími daty. Pro veškeré vizualizační metody umožňované Kibanou (mimo prvků Canvasu) viz obrázek 3.3.[8]



Obrázek 3.3: Nabídka vizualizačních prvků Kibany

Při vytváření grafů se vždy v levé části okna nachází ovládací nástroje a napravo se objevuje momentální vizualizace. Při změně jakéhokoliv parametru je nutné vizualizaci aktualizovat tlačítkem v pravém horním rohu levého okna. Vizualizace je nutné ukládat ručně.

Jak již bylo zmíněno v sekci 3.1.1, většina grafů potřebuje pro své správné vykreslení nastavit minimálně jednu agregaci typu Metrics (pro osu Y nebo určení počtu) a jednu agregaci typu Buckets (pro osu X nebo určení skupin dat). Tyto agregace se určují v levé části okna pod záložkou Data. Pro některé z agregací je nutné nastavit další parametry (například pro agregaci Terms pole, pro něž se mají data rozdělovat, nebo pro agregaci Percentiles jednotlivá procenta), tyto další vlastnosti se nastavují pod toutéž záložkou. Větší počet agregací jednoho nebo druhého typu se používá zejména má-li graf ukazovat rozdíl mezi hodnotami (například chceme-li v grafu zobrazit rychlost dvou závodníků, nebo porovnat náklad a výdělek firmy).

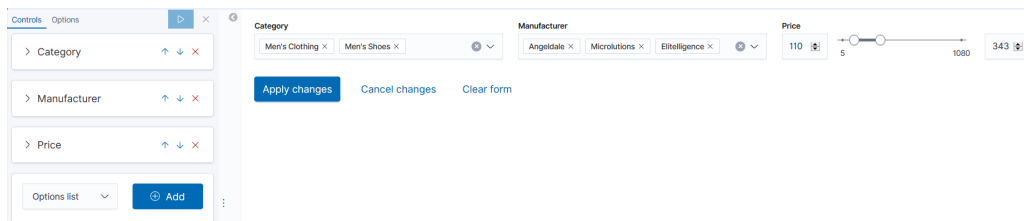
Grafy, jež mají osy, mají v levém ovládacím panelu dále záložku Metrics & Axes, kde je možné nastavit popisky a způsob zobrazení těchto os. Osy lze také zcela skrýt a nastavitu nich natočení popisků. Tytéž grafy mají pod záložkou Panel settings další dodatečné nastavení grafu, včetně zobrazování pomocných čar a umístění legendy.

Naopak grafy postrádající osu mají v těchto místech záložku Options, kde mají obdobné nastavení. Pod touto záložkou lze u některých grafů natáhnout natočení popisek nebo nastavení barevného modelu (barevný model není možné nastavit u každého typu). U jednoduchých grafů nezávisle na přítomnosti os není možné trvale skrýt legendu.

Ovládací panel

V momentální verzi Kibany je nabízen pouze jeden panel, který je věnován ovládacím prvkům a navíc i tento prvek je zatím pouze v experimentálním provozu. Jinak řečeno tento prvek se bude nejspíše dále rozvíjet nebo bude naopak z nedostatku zájmu odstraněn. Panel je možné vidět na obrázku 3.5.

Tento panel může obsahovat výběrové seznamy pro filtrování jednotlivých hodnot pole nebo slider pro výběr rozsahu hodnot z pole. Oba z těchto filtrovacích elementů vyžadují pro svou funkcionalitu nejprve vybrat index, nad kterým budou dále pracovat a dále pole téhož indexu. Díky možnosti výběru různých indexů je možné vytvořit ovládací panely schopné filtrovat na sobě nezávislá data a provést hloubkovou analýzu různých skupin (například zisky dvou různých firem vedených pod různými indexy). Pro jednotlivé prvky je vhodné doplnit popisek, jímž jsou prezentovány nejen při vytváření, ale i v náhledu.



Obrázek 3.4: Ovládací prvky vytvořené v Kibaně

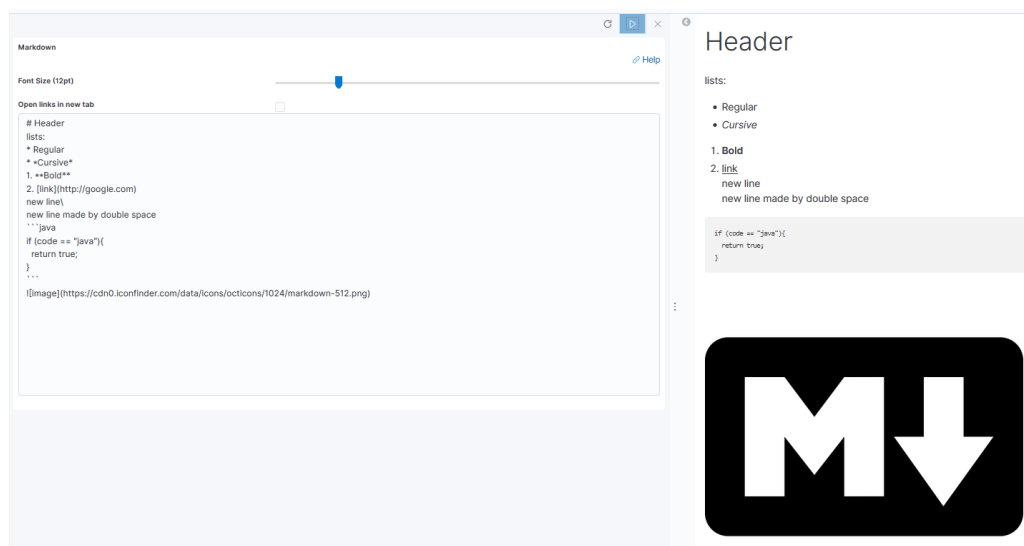
Pro slider rozsahu je nutné nastavit velikost kroku a počet desetinných míst, na něž má být výsledek zaokrouhlován. Výběrový seznam naopak umožňuje nastavit možnost více odpovědí a dynamického doplňování vstupu (je možné nejenom vybrat prvek z nabídky, ale i jej napsat manuálně). Seznamu může být přiřazen rodič, který dále filtruje možnosti listu. V našem případě má seznam Manufacturer nastavený seznam Category jako rodiče, tudíž vždy je možné vybrat výrobce jenom pro danou kategorii zboží.

Pod záložkou Options se nachází další dodatečné nastavení celého panelu. Je zde možné omezit působení defaultního časového filtru aplikace, dovolit aplikaci filtrů panelu nad celou databází Elastic Stacku. Uživatelé mohou

také povolit aplikaci filtrů při každé změně, v takovém případě z náhledu zmizí tlačítka pro potvrzení, zrušení a vyčištění filtrů.

Panel textu

Nástroj pro text, takzvaný Markdown, z důvodu jednoduchosti tohoto prvku vyžaduje pouze nastavení velikosti písma a samotný vstup. Využívaný odlehčený značkovací jazyk nedosahuje plného rozsahu Markdownu, přesto je většina nejdůležitějších funkcionalit zachována. Mimo funkcionalit viditelných na obrázku 3.5 je možné také vytvořit tabulku a aplikovat citace. Ale jak bylo dříve řečeno, není zachována veškerá funkcionalita Markdownu, pro text není možné nastavit horní ani dolní indexy nebo nastavit podtržení. Velikost obrázků není možné měnit a ani u částí kódu není možné využít podbarvení syntaxe. Nastavení textu zcela postrádá možnost změny barvy nebo fontu a není možné doplnit text daty z databáze.

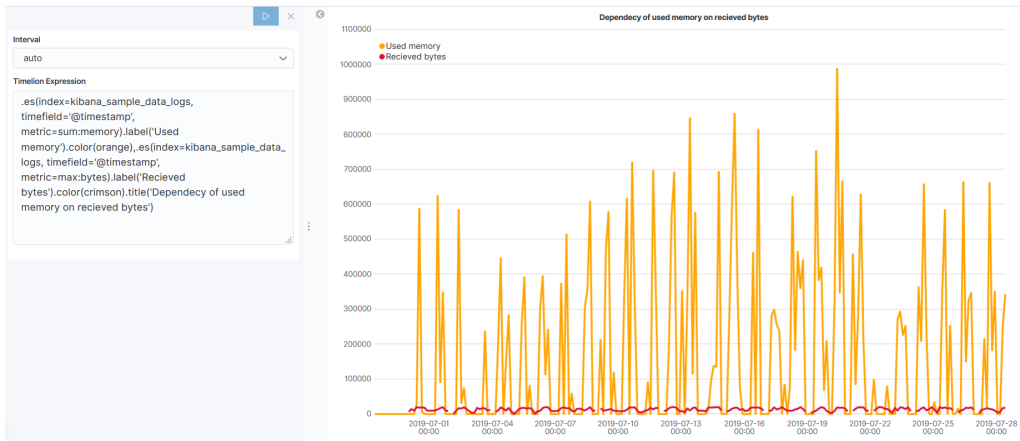


Obrázek 3.5: Markdown Kibany

Timelion

Timelion je nástroj Kibany určený k provádění vizualizací závislých na čase. Ačkoliv podobného efektu lze dosáhnout i při využití ostatních prvků Kibany, Timelion nabízí velkou škálu příkazů pro maximalizování efektivity a rozsahu této technologie. Díky tomu se stal jedním z nejvíce používaných vizuálních prvků Kibany.

Vizualizační element Timelion umožňuje upravovat vytvářený graf psaním Timelion výrazů, jimiž je možné nejen vybírat a filtrovat jednotlivá



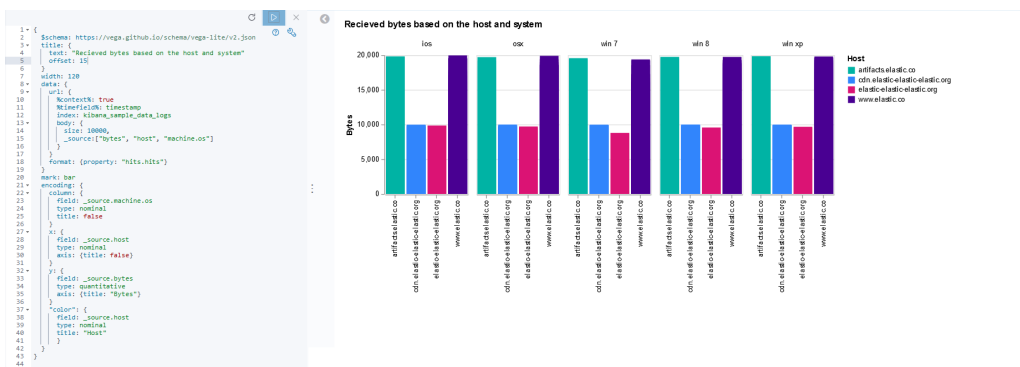
Obrázek 3.6: Vizualizace Timelion

data, ale také měnit typ grafu, barvu jednotlivých závislostí a legendu. Příklad Timelion vizualizace je vidět na obrázku 3.6.

Vega

Jak bylo dříve uvedeno, Vega gramatika je skriptovací jazyk sloužící pro tvorbu vizuálních prvků. Jinak řečeno, pokud uživatel potřebuje jakýkoliv typ grafu, může si jej vytvořit za pomoci Vegy sám, nebo prozkoumat její oficiální stránky a najít příslušný typ grafu. V momentální verzi Kibany je tato vizualizační metoda pouze experimentální, v budoucnu může být odstraněna.

Uživatel může k tvorbě elementů použít základní verzi Vegy nebo její zjednodušenou verzi Vega-Lite. Kibana podporuje obě z těchto verzí. Skriptovací jazyk je podporován v plném rozsahu, jsou tedy umožněny i interaktivní typy grafů.



Obrázek 3.7: Jednoduchá Vega gramatika

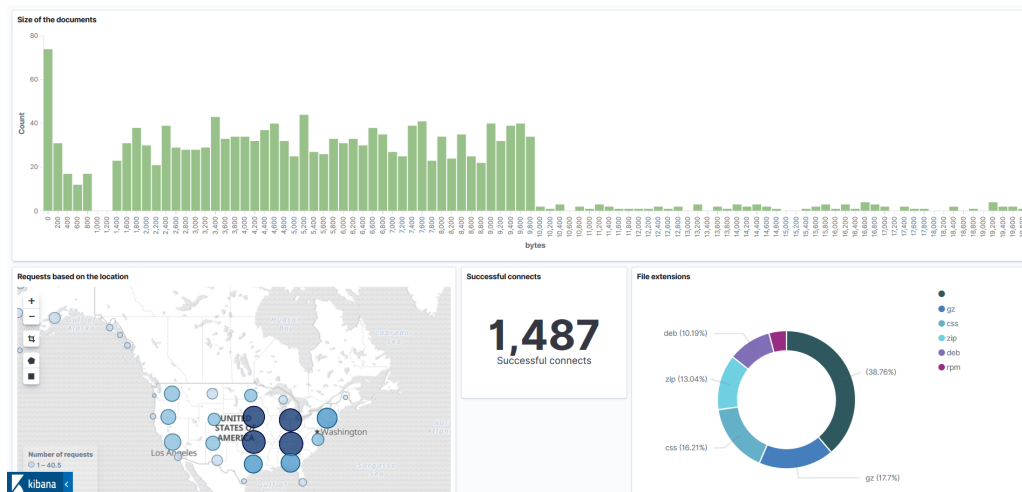
U grafů tvořených Vega gramatikou je přímo v kódu elementu možné určit, zda pro něj budou aplikovány zbylé prvky aplikace, či nikoliv.

3.1.3 Vizualizační skupiny

Ve většině případů jednotlivé vizualizační prvky nemusí být postačující, proto je nástrojem Kibana možno jednotlivé grafy kombinovat do panelů (Dashboard) nebo pláten (Canvas). Vhodnost jednotlivých komponent se liší podle momentální situace. Metoda Dashboard slouží pouze jako nadstavba nativních vizualizací Kibany, zatímco Canvas si vytváří vlastní vizualizace.

Dashboard

Prakticky od začátku své existence umožňuje Kibana vizualizaci dat prostřednictvím panelů nazývaných Dashboard. Jak již název napovídá, jedná se o pracovní plochy zobrazované pod doménou Kibany. Tyto plochy je možné rozdělit do několika menších částí, v nichž lze zobrazit jednotlivé grafy. Díky tomuto prostředku mohou být podobné nebo na sobě závislé grafy zobrazeny vedle sebe, což umožňuje jednodušší orientaci v zobrazovaných datech. Pro lepší názornost je na obrázku 3.8 ukázána skupina vizualizací Dashboard.



Obrázek 3.8: Jednoduchá skupina Dashboard

Počet ploch, jejich velikost ani jejich umístění není aplikací omezeno, je tudíž možné vytvořit několik různých Dashboard k snadnější analýze rozdílných dat. Pro větší názornost uvedu příklad. Předpokládejme, že firma vlastní databázi uživatelů firmou vyvíjených softwarů s informacemi o pořízení jednotlivých produktů, uživatelském hodnocení, atd... V aplikaci Kibana je teda vhodné vytvořit si dvě na sobě nezávislé plochy. V jedné z ploch

si firma může zobrazit grafy týkající se jednotlivých produktů, vztah mezi cenou a popularitou, popularitu produktů v jednotlivých uživatelských skupinách. Takto se může firma lépe zacílit na oblíbené produkty a nabízet lépe zacílené reklamy, tudíž více profitovat. V druhé ploše by tehdy mohli být zobrazeny data blíže se týkající konkurenceschopnosti firmy a jejího celkového výdělku v závislosti na čase. Takto připravená plocha umožňuje lépe si rozvrhnout rozložení kapitálu v jednotlivých sektorech firmy a lépe se připravit na propad zisku.

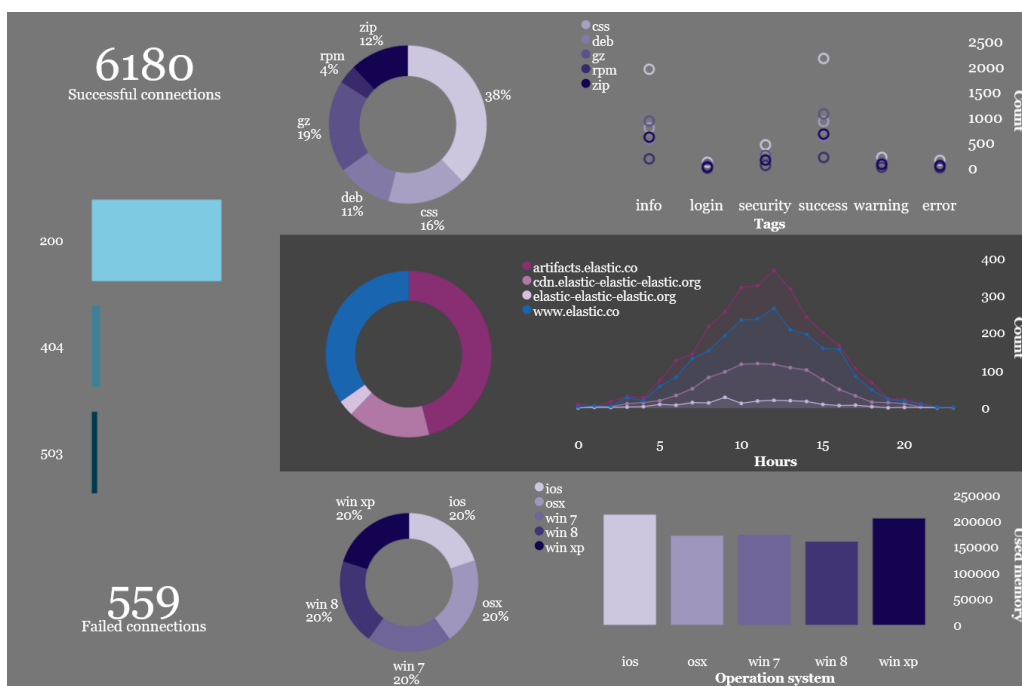
Každý Dashboard je skládán z jednotlivých dílčích vizualizací, které byli více rozepsány v předchozí sekci 3.1.2. V této sekci byly zmíněny ovládací prvky a vestavěné filtry grafů. Obě z těchto vlastností dosahují maximálního potencionálu až při začlenění do skupiny Dashboard, v níž jednotlivé nastavené filtry ovlivňují veškeré zbylé panely a grafy skupiny. Díky tomu jsou jednotlivé vizualizace více interaktivní a přehledné, tudíž prováděné analýzy dat dosahují větší efektivity. Pro každý z panelů je vhodné vytvořit adekvátní popis, který je automaticky zobrazen v horní části každého z panelů. Tento popis není možné skrýt.

Metodou Dashboard je možné dosáhnout efektivních vizualizací bez investování přílišného času. Vytváření ploch je jednoduché a nevyžaduje žádné speciální znalosti nebo zkušenosti, pěkné vizualizace mohou být vytvořeny i bez většího estetického cítění. Navíc prvky Dashboardu mají ve většině případů vestavěnou interaktivitu již v samotném grafu, je tudíž možné filtrovat výsledky kliknutím přímo do legendy nebo požadované části dat. Dodatečné filtry (viz 3.1.2) je také možné přidat jako samostatný panel.

Canvas

Hlavním rozdílem mezi vizualizací metodou Canvas a předchozí metodou je větší přizpůsobení prvků. K jednotlivým vizualizovaným prvkům se nepřístupuje jako k ohraničeným dlaždicím, spíše jsou uvažovány jako samostatné vrstvy Canvasu. V praxi to znamená, že vizualizace se mohou překrývat nebo kombinovat, což umožňuje kreativnější přístup.

Na rozdíl od skupiny vizualizací Dashboard, Canvas je spíše určen pro výrobu časově nezávislých prezentací, než hloubkové analýze dat. Jako taková technologie, Canvas rozděluje jednotlivá okna prezentace do stránek, které je možné v placené verzi exportovat do formátu PDF. Ve verzi 7.5.1 byla navíc nově přidána možnost vyexportovat Canvas pro webové stránky, tato metoda ale působí pouze staticky, jakákoliv nová data nebudou přidána do vizualizací, navíc veškeré filtrovací interaktivní prvky budou automaticky odstraněny. V momentálním vydání aplikace není možné měnit přechod mezi



Obrázek 3.9: Vizualizace v Canvasu

jednotlivými stránkami prezentace, ani nastavit automatické přechody. Příklad prvních dvou stran Canvasu je možné vidět na obrázcích 3.9 a 3.10.

U dříve zmíněných vizualizací Kibany byly zmíněny vestavěné filtry jednotlivých grafů, tato funkcionalita není bohužel v Canvasu nabízena, uživatelé si musí vystačit pouze s předem vyrobenými filtrovacími nástroji. Jistou omezenou interaktivitu je možné dodat za úpravy css jednotlivých prvků, to je bohužel omezeno na celé prvky. Díky tomu není možné upravovat zobrazené části grafu jinak než přes již vyrobené filtry Canvasu. Takové prvky je možné vidět na obrázku 3.10. Na nabízené filtry je také možné aplikovat css a částečně upravit jejich vzhled. Nicméně tyto filtry jsou pouze experimentálním nástrojem, nefungují tudíž vždy zcela spolehlivě a při vložení většího množství filtrů je mezi jednotlivými ovládacími prvky pouze logický operand AND. V novějších verzích má být možné aplikovat na filtry i operaci OR, bude tudíž možné zobrazit data s několika různými povolenými hodnotami.

Canvas také umožňuje snadné nastavení barevné palety jednotlivých elementů, pokud žádná z palet nevyhovuje, je možné nastavit vlastní barvy formou modelů HEX, RGB nebo HTML uznávaným jménem. Většina uživatelů ocení i možnost nastavit si barvu a velikost plátna. Vizualizace lze dále ozvláštnit přidáním barevných tvarů nebo obrázků, což nebylo dříve zmíněnou metodou možné. Novým prvkem je také možnost úpravy fontu a velikosti písma. Canvas není nutné ukládat jako Dashboard, ale automaticky

zaznamenává všechny změny sám.



| | |
|--------------------------|-------------|
| Host: | -- ANY -- |
| Extension: | -- ANY -- |
| Response: | -- ANY -- |
| Operating system: | -- ANY -- |
| Time range: | Last 1 year |

Obrázek 3.10: Filtrovací prvky Canvasu

Jak jsem již dříve zmínil, základní vizualizace z Kibany není možné importovat, prvky Canvasu je nutné vytvářet přímo v tomto prostředí. Při vytváření prvků se nejprve vybere typ vizualizace, jedná-li se o některý z jednodušších prvků (tvar, obrázek), stačí pouze nastavit jednotlivé vlastnosti v pravém panelu. U tvarů lze také oříznout rohy přidáním ovladače Container style z téhož panelu. Container style se nachází pod položkou Element style. Pod toutéž položkou se nachází i nástroj na úpravu stylu objektu přes CSS.

Složitější vizualizační prvky, jako například grafy, vyžadují nejprve zvolit zdroj dat. V momentální verzi programu jsou nabízeny čtyři datové zdroje a nastaví se opět v panelu na pravé straně, nejprve je ale nutné přepnout si záložku z Display na Data.

První z nabízených zdrojů je Timelion, tento zdroj je vhodný pro zobrazování časově závislých dat, například nárůst kapitálu firmy. Obdobně jako u filtrů Elasticsearch, Timelion využívá Timelion výrazy k filtrování jednotlivých dat, ale mimo to je ještě nutné nastavit interval, pro který bude daný graf vykreslován. Je-li v Timelion dotazu definovaná barva, bude tato část kódu ignorována, respektive přepsána nastavením prvku přes samotný Canvas.

Druhou možností je využití demo dat. Tato možnost je u prvků vybírána automaticky a pro reálné vizualizace není nikterak praktická. Jako takový slouží tudíž tento zdroj dat pouze k prvotnímu pochopení možností manipulace s daným typem grafu. Pracuje-li uživatel s Canvasem prvně, je nicméně vhodné ponechat toto nastavení než uživatel zcela pochopí jednotlivá nastavení grafu.

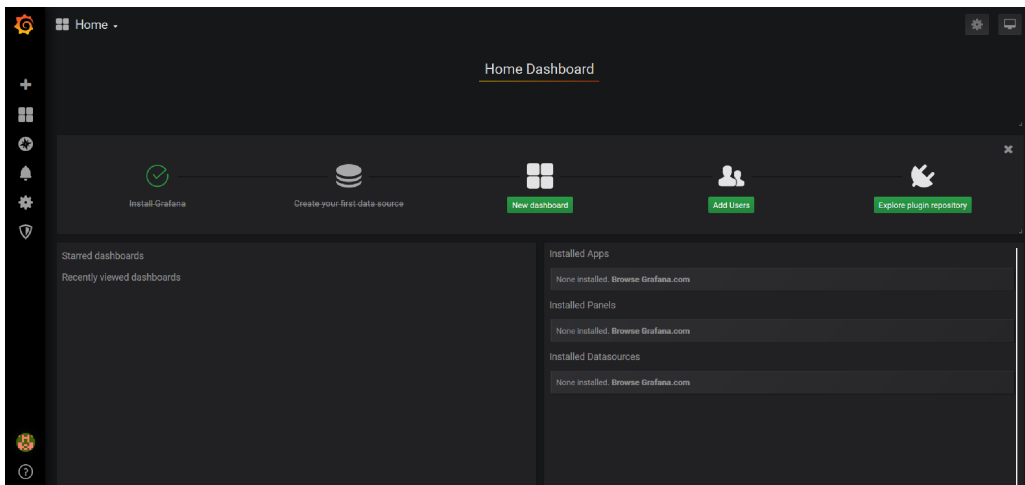
Třetí možností je připojit ke grafu data za využití Elasticsearch SQL query. Díky jednoduchosti a rozšířenosti jazyka SQL je pro většinu uživatelů snadné najít touto metodou potřebná data. Výsledky dotazu lze před

zpracováním nejprve zobrazit, čímž si lze nejenom zkontrolovat správnost SQL dotazu, ale i velikost množiny dat a její vhodnost zobrazení daným grafem. S dostatečnou znalostí jazyka SQL je možné vybrat data z daného intervalu a seřadit je podle některého z polí databáze.

Poslední nabízený zdroj dat je využít přímo hrubá data z ElasticSearch. Zde je nutné nejprve nastavit index, pod kterým jsou jednotlivá data uložena a pak množinu dat dále upřesnit pomocí Lucene query nebo přímo vybrat jednotlivá pole dat. Tato metoda umožňuje seřazení dat rovnou při jejich načítání do Canvasu.

3.2 Grafana

Grafana je vizualizační a analytický nástroj, obdobně jako prvky Elastic Stacku se jedná o aplikaci běžící na serveru. Grafana, stejně jako dříve zmíněné technologie, je multiplatformní, je podporovaná na většině operačních systémů. Tuto technologii je možné použít pro různé datové zdroje. Nástroj Grafana podléhá open source licenci.[5]



Obrázek 3.11: Úvodní okno Grafany

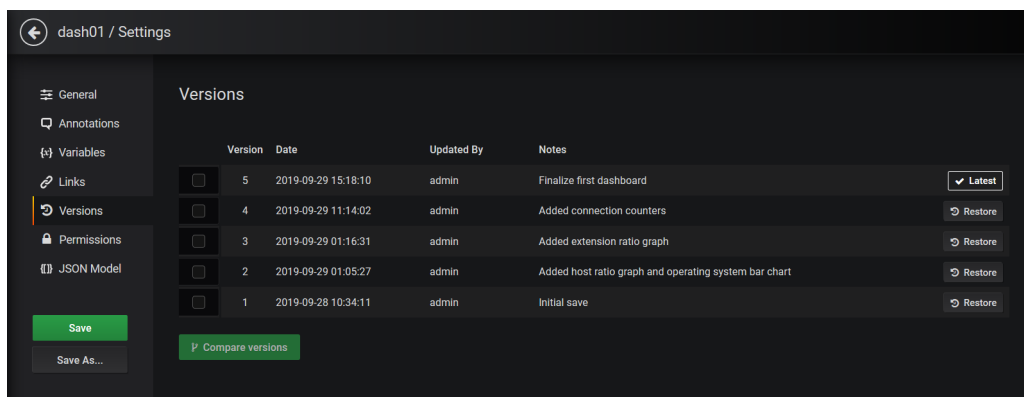
Aplikace se od dříve zmíněné Kibany liší už více uživatelsky přívětivým prostředím, pro uživatele je mnohem jednodušší upravit si Grafanu svým potřebám. Po přihlášení je uživatel přivítán úvodní stránkou (viz 3.11), na které vidí přehled svých projektů a postup pro další práci s nimi. Pro přiblížení přehlednosti aplikace ji porovnáme s nástrojem Kibana v následujícím příkladu. Obě aplikace umožňují předvolit si, motiv aplikace, ale zatímco v Grafaně je tato volba pod základním nastavením, Kibana tuto možnost

umístila do pokročilých voleb a velké množství uživatelů tudíž ani neví, že si mohou mód změnit.

Další příjemnou nadstavbou je systém upozornění, který je schopen při překročení předem stanovené hodnoty odeslat uživateli zprávu prostřednictvím podporovaných aplikací (Slack, PagerDuty, email atd.). K datům a jejich vizualizacím je možné přiřadit anotace.[6]

3.2.1 Systém vizualizací

Nástroj Grafana vykresluje jednotlivé vizualizace formou panelů (Dashboard), jednotlivé panely jsou vytvářeny přímo pod stejnojmennou záložkou. Přesný počet a velikost panelů nejsou opět omezeny, pokud to uživateli vyhovuje, může si pro každou vizualizaci vytvořit i samostatný Dashboard.

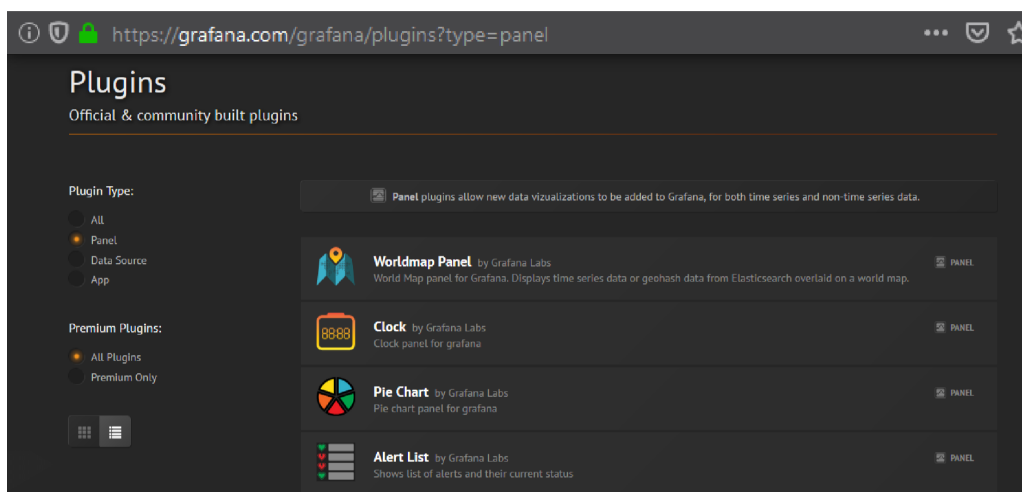


Obrázek 3.12: Verzování v aplikaci Grafana

Panely nejsou ukládány automaticky, ale při pokusu o opuštění stránky bez uložení je uživatel upozorněn, že změny budou ztraceny. Ukládané vizualizace jsou verzovány (viz 3.12), tudíž není možné je omylem znehodnotit. Pro snadnější orientování je zde možnost zobrazit změny, které byly provedeny v jednotlivých verzích vizualizace. Uživatelé mohou sdílet celý Dashboard i jednotlivé vizualizace, Dashboard je možné exportovat do formátu JSON. Pro veřejné prezentace je možné vytvořit odkaz na takzvaný Snapshot, kde jsou skryta veškerá citlivá data (struktury vizualizací, metriky, atd.). Panely je možné exportovat do formátu PNG. Podařené vizualizace je možné sdílet na stránkách jako šablony, nebo naopak neobratní uživatelé mohou využít již hotových Dashboardů.

Grafana po instalaci nabízí vcelku omezený výběr vizualizací, bez dalších úprav v aplikaci nalezneme pouze zcela základní grafy. Nicméně mnoho

dalších grafů je možné najít na oficiálních stránkách Grafany (stránka je ukázána na obrázku 3.13) a jejich instalace je velice jednoduchá. Pokud uživatelům nestačí ani tento výběr, má možnost napsat si vlastní plugin, který může být sdílen pro další uživatele. Programovací jazyk pro jednotlivé pluginy je zcela na uživateli. Po nainstalování pluginů je nutné restartovat server Grafany.

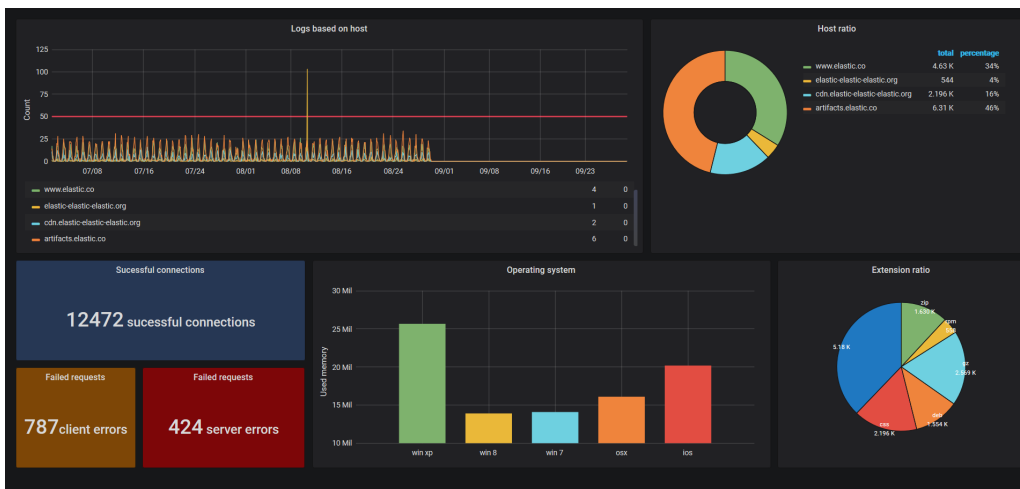


Obrázek 3.13: Nabízené pluginy Grafany

Jednotlivé panely mají jednotný postup tvorby, uživatelé jsou provedeni tvorbou panelu za pomoci záložek. Pod první záložkou se za pomoci Lucene query omezí volaná data, určí se metric agregace, jež má být nad jednotlivými buckety aplikována (agregace jsou totožné s agregacemi Kibany zmíněnými v 3.1.1) a data se rozdělí do skupin pomocí bucket agregací (taktéž totožné s agregacemi Kibany). Některé typy grafů vyžadují pro své správné vykreslení rozdělit data do bucketů podle času, i když jej nevyužívají. Jedná se například o koláčový graf nebo sloupcový graf.

Dále se pod záložkou Vizualization vybere typ grafu a specifikuje se vzhled os a legendy. Pod touto záložkou se nachází také nastavování prahových hodnoty a časových úseků. Poslední běžná záložka slouží k přiřazení názvu panelu a popisku, obě tyto možnosti jsou zcela volitelné. Zde je také možné nastavit průhlednost panelu (respektive zda má být ohraničen) a opakování dat. Některé grafy obsahují ještě jednu záložku umožňující nastavit již výše zmíněné alarmy při překročení hraničních hodnot.

Na rozdíl od Kibany, Grafana umožňuje pozměnit nejen barevné složky spektra, ale i průhlednost. Nicméně ovládání těchto změn není přehledné, u většiny grafů se příslušné okno zavolá kliknutím na barevný proužek v legendě. Tento přístup navíc není jednotný, některé grafy (například Sin-



Obrázek 3.14: Dashboard aplikace Grafana

glestat), mají toto nastavení přímo pod záložkou Vizualization při své tvorbě. Na obrázku 3.14 je vidět jednoduchý Dashboard pro větší názornost.

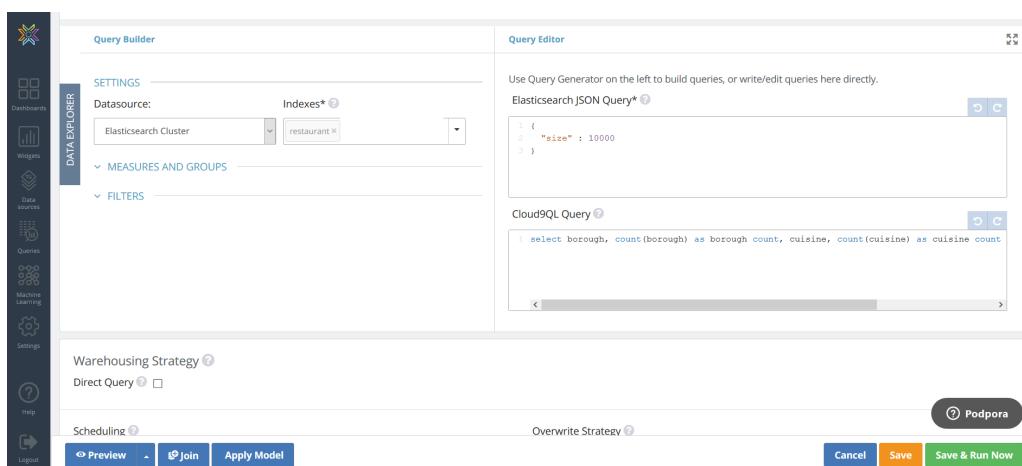
Filtrování v Dashboardu lze docílit kliknutím na data v grafu nebo jeho legendě, toto řešení nicméně filtruje data v daném panelu. K filtrování výsledků napříč celým Dashboardem je nutné vytvořit proměnné, tyto ovládací prvky jsou reprezentovány dropdown listem nad panely a samostatné proměnné musí být začleněny do grafů již při jejich tvorbě.[6]

3.3 Knowi

Knowi je placená webová aplikace sloužící k vizualizování a analýze dat z různých datových zdrojů. Knowi nabízí tři druhy licencí: Basic, Team a Enterprise. Samotné ceny těchto tří tříd nejsou obecně známé, cena je dojednána až podle specifikovaných požadavků (velikost týmu, počet dat, atd...). Aplikace nabízí 21 denní trial.[9]

Aplikaci není nutné instalovat, vše probíhá na vnějším serveru, což bohužel znatelně zpomaluje funkčnost aplikace. Knowi podporuje různé typy datových zdrojů, jmenovitě jsou to NoSQL zdroje (například MongoDB, Elasticsearch, InfluxDB a Hive), klasické SQL databáze (například MySQL, PostgreSQL a Oracle), datová úložiště (Amazon Redshift, Snowflake a Teradata) a načítání ze souborů (podporovaná je většina tabulkových formátů). Datové zdroje je možné v dashboardech kombinovat.

Vizualizace (tzv. widgets) je možné vytvořit třemi rozdílnými způsoby. Ve většině případů je vhodné vytvářet je za použití speciálních dotazů nativních pro používané datové zdroje. Příklad takového dotazu pro Elastic-



Obrázek 3.15: Elasticsearch query v Knowi

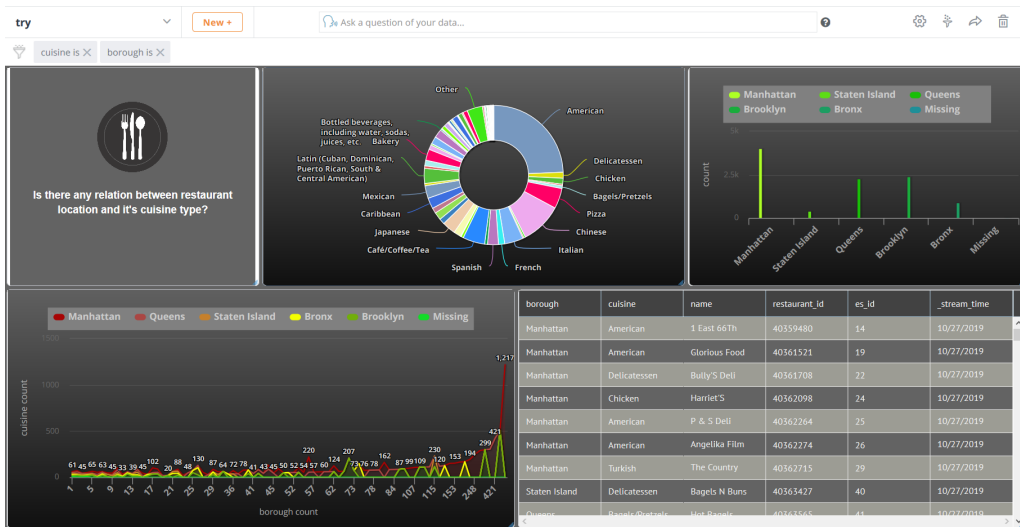
Search je možné vidět na obrázku 3.15. Jelikož aplikace nenabízí u většiny widgetů agregace, data je nutné získat z Elasticsearch již ve zpracovaném formátu. Před uložením je vhodné si používaná data zobrazit za pomoci preview v levém dolním rohu aplikace. Při uložení dotazu je automaticky generován widget nad danými daty, ale uživatelé by si jej vždy měli upravit vlastním potřebám (změnit typ grafu, aplikovat filtry, doplnit popisky, nastavit barvy elementů a pozadí, atd...).

Widgety je možné vytvořit taky duplikací a úpravou již vzniklého widgetu, to je vhodné zejména potřebujeme-li dva grafy nad stejnými dotazy. V neposlední řadě je možné generovat widgety aplikací dotazů přes pole nad dashboardem. Tyto dotazy se provádějí přímo v anglickém jazyku.[10]

Typově nabízí aplikace Knowi největší počet vizualizačních prostředků z dosud testovaných nástrojů. Knowi nabízí nejenom obvyklé grafy jako sloupcový a koláčový, ale i méně časté box ploty, Sankeovy diagramy a dokonce je nabízena možnost aplikovat heatmappy nad vlastními obrázky. Navíc aplikace obsahuje u většiny elementů (včetně jednotlivých typů widgetů) podrobný popis s návodem na použití.

Jak již bylo dříve naznačeno, jednotlivé widgety mohou být opět kombinovány do dashboardů. Vizualizace jsou přidávány přetažením vybraných položek ze záložky Widgets do plochy dashboardu. Uživatelé mohou u dashboardů nastavit barvu pozadí a upravit některé vlastnosti vložených widgetů (například zda má být zobrazován jejich rámeček a titulek), jiné vlastnosti zůstávají však zcela neměnné. Filtry v jednotlivých elementech nejsou aplikovány automaticky na všechny prvky dashboardu, ale obdobně jako v aplikaci Grafana zmíněné v sekci 3.2.1 je možné vytvořit filtry prostřednictvím dropdown listů. Příklad vytvořeného dashboardu je přiložen na

obrázku 3.16.



Obrázek 3.16: Dashboard aplikace Knowi

Knowi umožňuje exportovat dashboardy do formátu PDF, sdílet je prostřednictvím emailu nebo odkazem. Aplikace vytváří i HTML tagy pro přímé vložení do webové stránky. Před samotným exportem je vhodné nastavit, jaká práva budou návštěvníci panelu mít.

3.4 Jazyk R

R je programovací a vizualizační jazyk vycházející z jazyka S. Tento jazyk je nekomerční a je jej možné spustit nezávisle na platformě. R se využívá zejména k statistickým a analytickým rozborům dat. Hlavním vývojovým prostředím jazyka je RStudio.[15]

Jazyk R podporuje rozsáhlý počet zdrojů dat, mezi těmito zdroji i data uložená v aplikaci Elasticsearch. K připojení těchto dat k programu v R je nutné nainstalovat elastic knihovnu. Po nainstalování a připojení dané knihovny je nejprve nutné připojit vytvářený projekt k používané databázi za použití příkazu `connect`, pak je již možné načítat jednotlivá data za příkazů `Search` nebo `docs_get`. Jednotlivé parametry pro specifikaci dat a indexů jsou pro oba příkazy prakticky totožné. Tento postup se používá ale spíše k ovládní Elasticsearch, data jsou získávána ve formátu JSON, což komplikuje jejich další vizualizaci.[13]

Pokud vlastní uživatelé na Elasticsearch platinum verzi (placená verze) a nainstalují si v něm JDBC driver, je možné pomocí knihovny RJDBC v R tento driver připojit. Tímto způsobem je pak možné zcela jednoduše

tvořit požadované grafy. Pokud je R aplikace spouštěna na serveru a data jsou zadána tímto způsobem (nebo je-li zdroj dat umístěn online), stránka je schopná aktualizovat své grafy při změnách dat.[17]



Obrázek 3.17: Dashboard jazyka R

Jednotlivé grafy se do prostředí načítají prostřednictvím knihoven, v případě neexistující nebo nevyhovující knihovny je ale ovšem možné vytvořit si zcela vlastní vizualizace. Většina knihoven umožňuje maximální kontrolu nad jednotlivými grafy, uživatelé mohou nastavit jednotlivé barvy prvků, popisky grafů, existenci i případný formát legendy. Z jednotlivých knihoven musím zmínit hlavně knihovnu flexdashboard, která sice nenabízí samotné grafy, ale umožňuje začlenění grafů z jiných knihoven do jednoho uceleného a interaktivního dashboardu. Dashboard vyrobený v R je vidět na obrázku 3.17.[15]

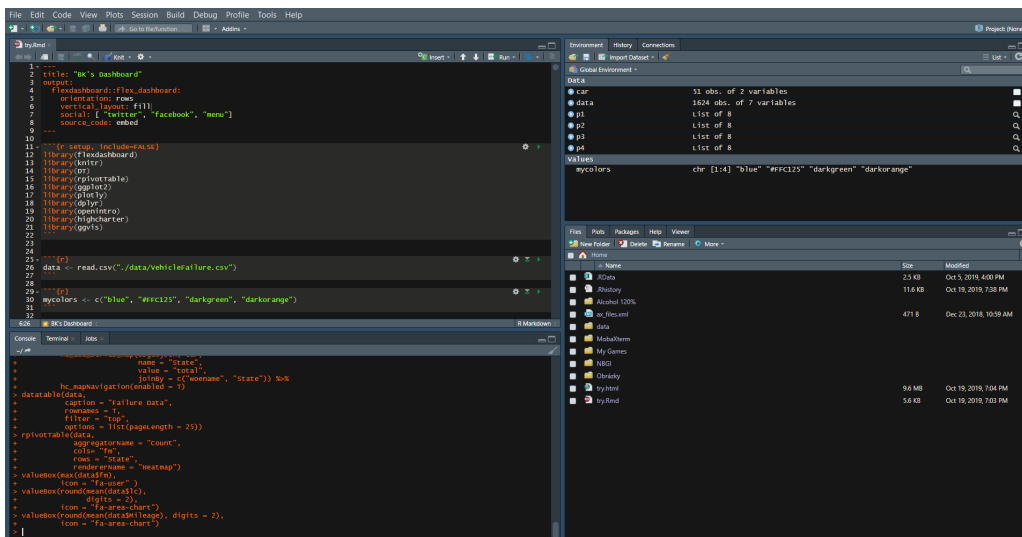
R také nabízí knihovny (např. knitr) pro export vytvářených vizualizací do různých formátů, ať již to jsou formáty statické (PDF) nebo i dynamické (HTML). Díky tomu je možné uchovat maximální interaktivitu programů bez nutnosti instalace překladače jazyka R na jednotlivá zařízení.[14]

Bohužel, jsou-li používány knihovny, do jednotlivých grafů není možné začlenit interaktivní filtry, jimiž by byl upraven celý dashboard. Pokud by si uživatel programoval celé grafy sám, nejspíše by to ale možné bylo. Nicméně je možné do dashboardu naimplementovat drop-down filtr, který je schopný přepočítat globálně všechny proměnné a překreslit grafy.

Jazyk R má oproti výše zmíněným programům tu výhodu, že umožňuje vytvářet prakticky libovolné vizualice a aplikace. Naopak je nutné počítat, že trvá déle se s jazykem seznámit, a i v případě pokročilých uživatelů, výroba interaktivních dashboardů zabere více času.

3.4.1 RStudio

RStudio je hlavní vývojové prostředí pro jazyk R. Prostředí RStudio je multiplatformní, je nabízené v placené i volné verzi. Ve volné verzi jsou některé vlastnosti omezeny, ale běžný uživatel tyto nedostatky příliš nepocítí. Jednotlivé knihovny pro vlastní kód je možné instalovat rovnou z aplikace respektive rovnou z kódu.



Obrázek 3.18: Prostředí RStudio

Jak je vidět na obrázku 3.18, prostředí RStudia je defaultně rozděleno do 4 částí. V levé horní části aplikace je psán samotný kód programu, napravo jsou dále zobrazovány použité proměnné a jejich hodnoty. V dolní části se pak nalevo nachází konzole s výpisy aplikace a terminál. V pravém dolním rohu zobrazuje aplikace obsah pracovního adresáře (defaultně dokumenty). Layout aplikace je samozřejmě možné upravovat, aplikace dokonce nabízí i změnu motivu (na obrázku je vidět tmavý motiv, ale aplikace defaultně používá motiv světlý).

Jednotlivé části kódu se píšou do částí (takzvané chunk) obklopených třemi zpětnými apostrofy. Prostředí RStudia umožňuje spouštění jednotlivých částí zvlášť, a po proběhnutí dané části kódu zobrazí vzniklý výstup. Zejména při vykreslování jednotlivých grafů je tato vlastnost velmi užitečná, je totiž možné ihned detekovat nepřesnosti nebo chyby v kódu.

Samozřejmostí je i našeptávání, jsou-li na začátku kódu načteny data, je možné dokonce referovat přímo sloupce databáze. Aplikace nabízí i refactoring a kontrolu pravopisu. V části 3.4 jsem zmínil možnost exportu programu do formátu HTML, pokud je importována knihovna knitr, export je možné

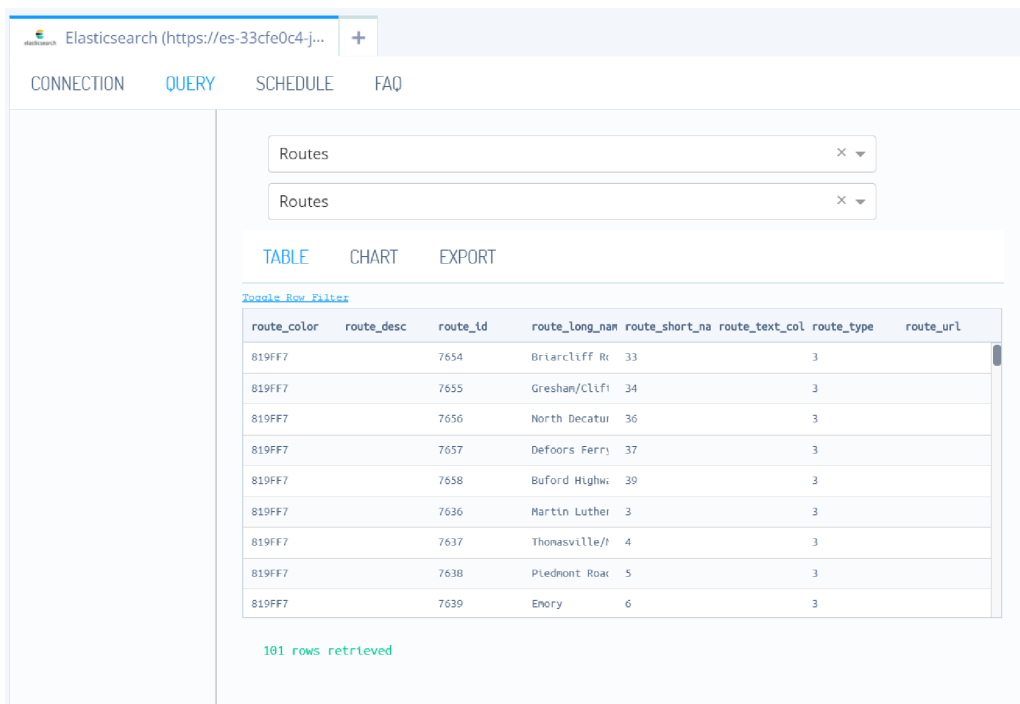
najít pod záložkou File, nebo pod ikonou pletacích jehlic v liště nad psaným kódem.

3.5 Plotly

Plotly je webová vizualizační aplikace založená na stejnojmenné knihovně jazyka Python. Aplikace je dostupná pod dvěma licencemi. Pod licencí Basic jsou nabízeny veškeré nástroje bez jakýchkoliv poplatků, jedinými omezeními jsou objem ukládaných dat a veřejnost vytvořených vizualizací. Placenná verze Enterprise naopak umožňuje vytváření privátních grafů a větší objem dat.

Samotná aplikace nevyžaduje instalaci, nicméně pokud uživatel nechce používat jako zdroj základní tabulkové formáty, ať již uložené online nebo na lokálním zařízení, je nutné využít nástroje Falcon. Tento nástroj již instalaci vyžaduje.[12]

3.5.1 Falcon



The screenshot shows the Falcon tool interface. At the top, there is a search bar with the text 'Routes' and a dropdown menu. Below the search bar, there are two tabs: 'TABLE' and 'CHART'. The 'TABLE' tab is selected, and a table of results is displayed. The table has the following columns: route_color, route_desc, route_id, route_long_name, route_short_name, route_text_color, route_type, and route_url. The results are as follows:

| route_color | route_desc | route_id | route_long_name | route_short_name | route_text_color | route_type | route_url |
|-------------|------------|----------------|-----------------|------------------|------------------|------------|-----------|
| 819FF7 | 7654 | Briarcliff R | 33 | | 3 | | |
| 819FF7 | 7655 | Gresham/Cliff | 34 | | 3 | | |
| 819FF7 | 7656 | North Decatur | 36 | | 3 | | |
| 819FF7 | 7657 | Defoors Ferry | 37 | | 3 | | |
| 819FF7 | 7658 | Buford Highway | 39 | | 3 | | |
| 819FF7 | 7636 | Martin Luther | 3 | | 3 | | |
| 819FF7 | 7637 | Thomasville/ | 4 | | 3 | | |
| 819FF7 | 7638 | Piedmont Road | 5 | | 3 | | |
| 819FF7 | 7639 | Enory | 6 | | 3 | | |

Below the table, it says '101 rows retrieved'.

Obrázek 3.19: Query v nástroji Falcon

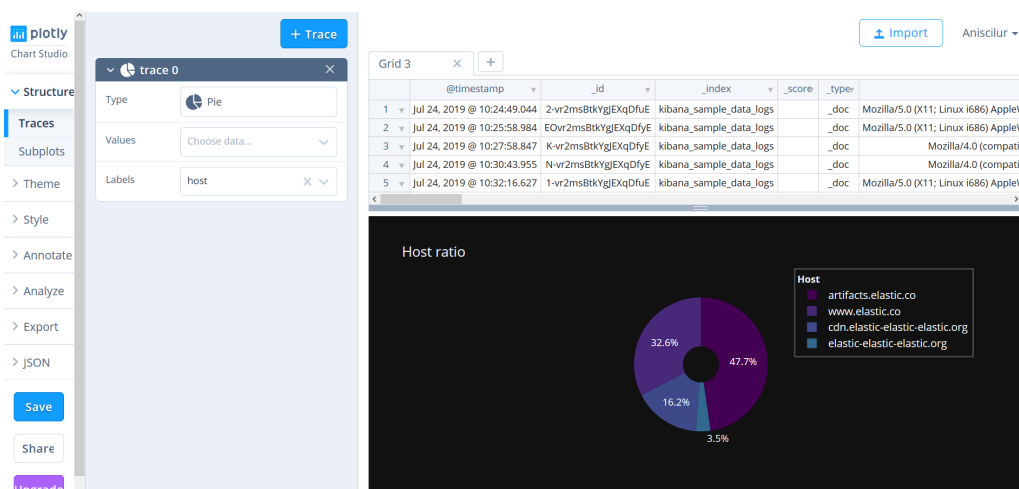
Nástroj Falcon slouží k importování datasetů z vnějších zdrojů (firemní nebo osobní databáze) do databáze Plotly. Tento nástroj je platformě ne-

závislý. Po instalaci je nutné vybrat typ databáze, z které mají být data načtena. Další nastavení závisí na vybraném datovém zdroji, ale obecně je možné říci, že je do okna nutno zadat adresu zdroje a přihlašovací údaje nad tímžte zdrojem. Po potvrzení zadaných údajů jsou údaje uloženy a zamčeny. Pro změnu zadaných údajů je nutné kliknout na tlačítko "Edit Credentials". Většina zdrojů nabízí testovací servery na něž je možné se připojit.

Po úspěšném připojení k Elasticsearch (či jinému zdroji) je možné zaslat data na server Plotly jako statická data prostřednictvím query (viz obrázek 3.19) nebo data posílat v pravidelných intervalech prostřednictvím scheduled query a udržovat je tak aktuální. Statická i dynamická data jsou po odeslání na server reprezentována tabulkou s daty. Příklad takových dat je vidět na obrázku 3.20. Data nejsou aplikací Plotly agregována, je nutné je načíst již v agregovaném formátu.

3.5.2 Prostředí Plotly

Každý přihlášený uživatel Plotly má pod svým profilem přehledně rozbrazeny všechny již vytvořené vizualizace a datové zdroje. Vlastněné soubory je navíc možné filtrovat podle typu i času přidání. Každý z vizualizačních elementů je možné zobrazit jako editor i jako normální uživatel.



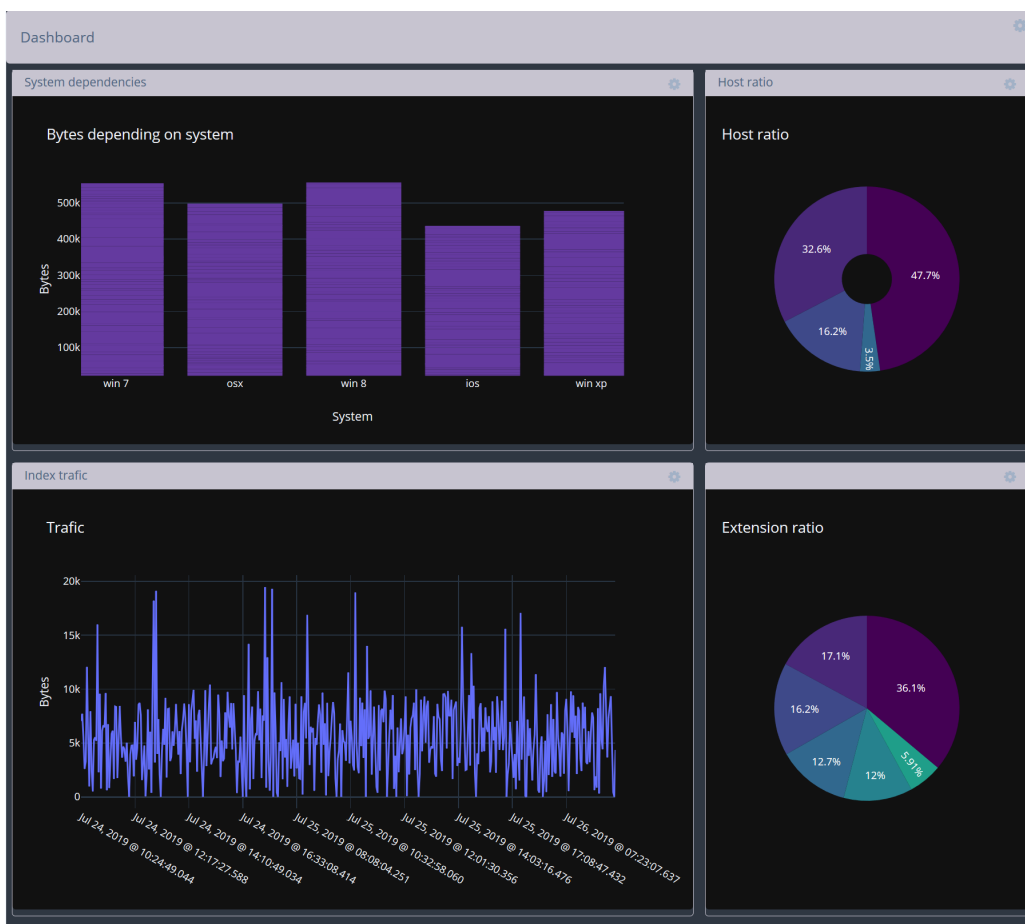
Obrázek 3.20: Jednoduchá vizualizace v Plotly

Jednotlivé grafy jsou vytvářeny přímo z datových zdrojů, sdílí s nimi tudíž okno (viz obrázek 3.20). Typ grafu je vybírán pod záložkou Structure, přesněji řečeno pod daným Trace reprezentujícím část dat. V současné verzi jsou podporovány klasické čárové, koláčové, sloupcové i bodové grafy. Mimo těchto grafů jsou aplikací nabízeny i heatmapy, histogramy, konturové his-

togramy, tabulky, box-ploty, treemapy, Sankeyovi diagramy a několik typů map.

U každého z grafů je možné nejenom nastavit barvy, jimiž je reprezentován, ale i barva pozadí a poloha legendy. Data mohou být dále anotována. Grafy nejsou automaticky ukládány ani verzovány, ale uživatel je při opuštění okna informován, pokud nebyly změny uloženy.

Jednotlivé grafy je možné kombinovat do dashboardů, u nichž je možné upravovat velikost a barvu jednotlivých prvků. Pod nastavením dashboardu se také nastavuje, zda uživatelé chtějí povolit filtrování dat a vyhledávání nad nimi. Příklad vytvořeného dashboardu je možné vidět na obrázku 3.21.



Obrázek 3.21: Dashboard aplikace Plotly

3.6 Aplikace nevyhovující zadání

Programy testované v této práci mají umožňovat vizualizace přímo nad aplikací ElasticSearch. Z důvodu úplnosti v této části zmíním nejen programy, které tuto podmínku nesplňují, ale i technologie, které byly zamítnuty kvůli značným nedostatkům ve funkčnosti.

Při prvotním průzkumu vyvolali zájem aplikace Prometheus, Losene a Logz.io, ale při důkladnějším testování bylo zjištěno, že tyto aplikace využívají k samotným vizualizacím jiné aplikace. Konkrétně Prometheus využívá Grafanu a Logz.io s Losene vizualizují prostřednictvím Kibany. Všechny tyto aplikace nabízejí vlastní vyhledávací enginy, ale nerozšiřují vlastní možnosti vizualizace, tudíž jsou pro účely této studie ze zřejmých důvodů nevyhovující.

V průběhu studie byly zvažovány i aplikace Geckoboard, Sumo Logic, Highcharts, Graphite, SenseLogs, Seq a jiné, ale tyto programy neumožňují vizualizovat nad daty ElasticSearch, proto byly i ony zamítnuty. Dále byly testovány aplikace Graylog, Datadog a Cyclotron, ale jejich funkčnost je uživatelsky nepřívětivá nebo postrádají nástroje pro hloubkové analýzy.

4 Výsledky porovnání

| Technologie | Licence | Interaktivita | Export |
|------------------|---|--|--|
| Kibana Dashboard | Basic (zdarma) Gold Platinum Enterprise | vestavěné filtry panelů speciální panel s dropdown listem a sliderem | embed iframe permalink PDF (od Gold) PNG (od Gold) |
| Kibana Canvas | | experimentální dropdown filtr | statický snippet PDF (od Gold) PNG (od Gold) |
| Grafana | Basic (zdarma) Enterprise | dropdown filtry (tzv. variables) | permalink snapshot (lokální i centralizovaný) link na obrázek |
| Knowi | Basic Team Enterprise | dropdown filtry | permalink PDF embed iframe email |
| Plotly | Basic (zdarma) Enterprise | vestavěné filtry panelů filtrování výběrem | permalink email embed iframe |
| R | Jazyk nekomerční X JDBC pod Elasticsearch vyžaduje Platinum | dropdown filtry (nutno doimplementovat) | PDF a HTML |

Obrázek 4.1: Porovnání technologií

Jak je patrné z obrázku 4.1, všechny testované technologie umožňují nějakou formu filtrování dat a jejich export. Nejvíce uživatelsky přívětivé filtrování poskytuje technologie Kibana Dashboard, kde je možné provádět filtrování dat přímo kliknutím na část grafu a data se upraví i ve zbytku vizualizace. Vizualizace v této aplikaci nenabízejí ale velké možnosti úprav, výsledné vizualizace i přes velkou přehlednost nepůsobí esteticky přívětivě. Tato technologie umožňuje i přímý export iframe pro vkládání do HTML stránky.

Přesným opakem je technologie Kibana Canvas, která neumožňuje žádný export dynamických dat, a nabízené interaktivní prvky nejsou zatím zcela využitelné. Implementovaný dropdown filtr nefunguje správně u každého

typu dat a není možné jednotlivé filtry nad jedním polem kombinovat. Vizualizace vytvářené touto technologií jsou nicméně kompletně přizpůsobivé, dokonce je možné jednotlivé vizualizace překrývat.

Grafy aplikace Knowi je možné filtrovat přes dropdown filtr obdobně jako Kibana Canvas, narozdíl od ní je možné ale jednotlivé filtry kombinovat. Dashboardy je možné sdílet prostřednictvím permalinků nebo z nich vytvořit iframe. Problémem této metody je ale absence samostatných agregací, data musí být načteny již agregované. Zároveň data jsou ukládána přímo pod serverem Knowi, což nemusí být vždy žádoucí. Tento program je jediným vybraným programem, jež neposkytuje nekomerční verzi, pouze krátký trial.

Aplikace Plotly nabízí několik metod filtrování, zejména je možné data filtrovat přímým kliknutím do grafu, nebo daná data manuálně vybrat z části datasetu. Dashboardy je taktéž možné sdílet permalinkem, nebo je vložit přímo do stránky vyexportovaným elementem. Bohužel, aplikace nevytváří agregace, data musí být načítána již agregována. Dalším problémem je, že aplikace neprovádí pouze dotazy, ale ukládá si veškerá data do vlastní databáze na svém serveru, což opět nemusí být vždy žádoucí.

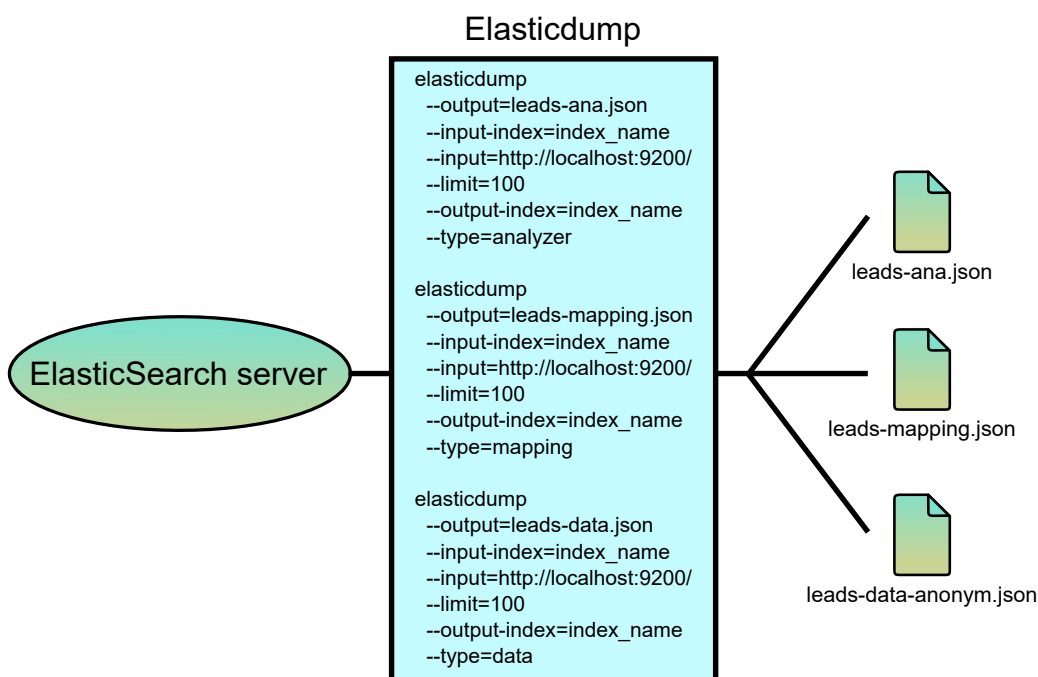
Vizualizace je možné vytvořit i ve vizualizačním jazyku R. Jelikož se nejedná o jedinou aplikaci, v této technologii je možné maximálně využít potenciálu vizualizací a reprezentovat libovolná data jakýmkoliv potřebným způsobem. Naopak nevýhodou je, větší režie tohoto přístupu, tvorba vizualizací trvá znatelně déle. Pro maximalizaci komunikace mezi R a Elastic Search je navíc nutné mít nainstalován JDBC, jež je vývojáři Elastic podporován pouze při vlastnění licence Platinum.

Aplikace Grafana kombinuje přístupy dříve zmíněných aplikací do jednoho celku. Přestože aplikace nenabízí filtrování veškerých vizualizací přímo kliknutím do grafu, je jednoduché nad grafy vytvořit plně funkční dropdown filtry. Vizualizace je možné více přizpůsobit než vizualizace Kibany Dashboard a režie na jejich vytváření není nikterak komplikovaná. Aplikace také nabízí export do obrázků, permalinků a snapshotů, čímž lze sdílet statická i dynamická data. Narozdíl od technologií Kibany, veškeré ze zmíněných funkcí jsou dostupné již při použití základní licence, což činí tento nástroj nejvhodnějším testovaným nástrojem pro zpracování této práce.

5 Testovací data

K hlubšímu průzkumu nástroje Grafana byla dodána skutečná firemní data ve formátu JSON. Tato data byla vyexportována z ElasticSearch serveru prostřednictvím nástroje Elasticdump, jež je schopný nejenom vyexportovat obsah serveru, ale i jeho nastavení a soubory k inicializaci indexu. Prostřednictvím téhož programu je možné data zálohovat, nebo ze vzniklých záloh opět obnovit stav serveru.

Elasticdump je platformově nezávislý nekomerční nástroj podporující většinu verzí serveru ElasticSearch. Nástroj je schopný pracovat nejen s přímými odkazy na server (kopírování dat z jednoho serveru do jiného serveru), ale i s lokálně uloženými soubory (výhodné zejména pro export a zálohování). Nástroj nabízí i možnost ukládání dat odpovídajícím specifickým dotazům json query.[3]



Obrázek 5.1: Export prostřednictvím Elasticdump

Pro účel této práce byly poskytnuty tři soubory a to leads-ana, leads-mapping a leads-data-anonym. Tyto soubory jsou exportovány výše zmíněným nástrojem, a neobsahují tudíž přirozené oddělovače jako nový řádek nebo tabulátor. Každý ze souborů slouží jinému účelu, první soubor se stará o inicializaci indexu, druhý soubor namapování datových typů a poslední

soubor obsahuje samotná data. Bližší informace o exportu těchto dat je vidět na obrázku 5.1, import by byl prováděn obdobným postupem, pouze by bylo nutné zaměnit položky input za output a obráceně.

Při prozkoumání jednotlivých souborů je možné zjistit bližší informace o zapůjčeném serveru. Soubor k inicializaci databáze obsahuje informace o počtu shardů (v našem případě pět) a replik (jedna). Dále je z téhož souboru možné vyčíst, že maximální počet navrácených dokumentů při dotazech byl navýšen ze základní hodnoty 10000 na 500000.

Soubor pro mapování datových typů obsahuje informace o typech jednotlivých polí databáze. Z námi obdržенých dat je patrné, že byl použit starší server, jelikož data obsahují zastaralý typ string. Jak již bylo dříve popsáno v části 2.1, tento typ byl ve verzi 5 nahrazen typy text a keyword. Ve verzi 6 byla poté odstraněna veškerá zpětná kompatibilita, díky tomu není již možné vytvářet indexy s těmito typy. Pro replikování uživatelského serveru je tudíž nutné využít starších verzí.

Poslední soubor reprezentující samotná data není možné bez porušení anonymity firmy hlouběji popisovat, nicméně se jedná o soubor velikost 250 MB. Z tohoto souboru je možné vyčíst, že datum přidání dokumentu do ElasticSearch se nazývá *createDate*.

Starší formát dat, respektive verze serveru ElasticSearch, není pro nástroj Grafana překážkou. Nástroj Grafana umožňuje výběr z jednotlivých verzí ElasticSearch, nicméně je nutné znát používanou verzi, jelikož nástroj nenabízí možnost autodetekce.

5.1 Připojení Grafany k ElasticSearch

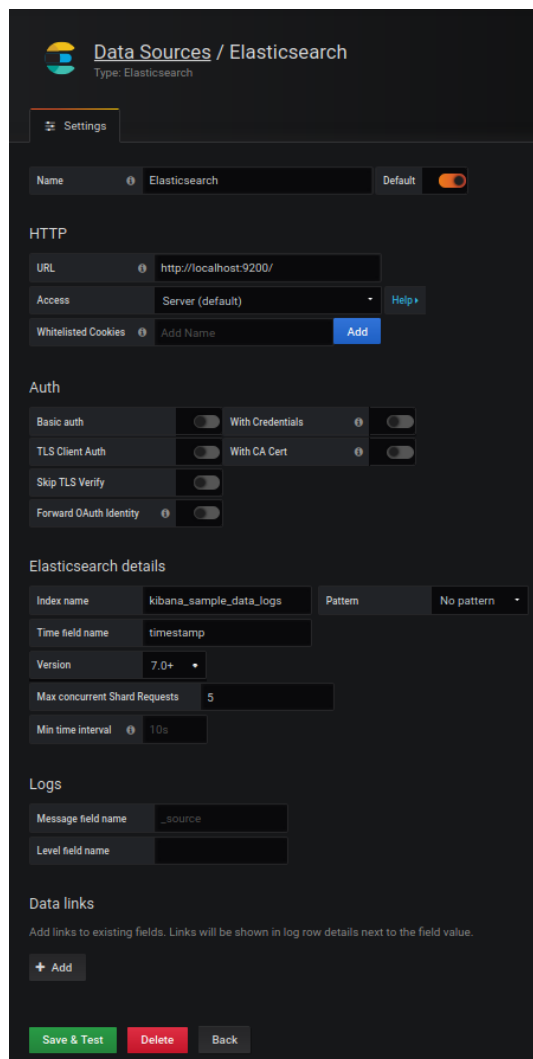
Grafana umožňuje připojit se k 14 různým zdrojům dat, pokud požadovaný zdroj není v základní nabídce, je možné doinstalovat si datový zdroj z dostupných pluginů. Nenachází-li se požadovaný zdroj ani mezi pluginy, je možné vytvořit si plugin vlastní, nebo požádat o jeho vytvoření. Z bezpečnostních důvodů mohou pouze uživatelé s rolí Admin vytvářet a upravovat datové zdroje.

ElasticSearch je jedním z nativních zdrojů aplikace, Grafana tudíž nabízí širokou škálu nastavení pro přizpůsobení spojení. V rámci aplikace je každý datový zdroj reprezentován svým jménem, pokud je zdroj nastaven jako defaultní zdroj, bude nastaven automaticky jako zdroj nově vytvářených vizualizací.

Grafana podporuje pouze zdroje dostupné za pomoci url, a ani ElasticSearch není výjimkou. Jelikož je většinou využíván lokální server, url má jako

nápovědu `http://localhost:9200/`, což je defaultní nastavení Elasticsearch serveru. Dále je možné nastavit, zda má být datový server dostupný přes server Grafany, nebo zda má být dostupný přímo z prohlížeče. Druhý případ může podléhat požadavkům CORS (tj. Cross-Origin Resource Sharing), z toho důvodu je preferována první možnost.

CORS je mechanismus vyžívající dodatečných HTTP hlaviček pro dodání webové aplikaci běžící na jedné doméně přístup ke zdrojům umístěných na jiné doméně. Z bezpečnostních důvodů je tento přístup ve většině prohlížečů omezován.[1]



Obrázek 5.2: Elasticsearch datový zdroj v Grafaně

Grafana u datového zdroje také počítá s možným zabezpečením Elasticsearch, proto obsahuje dodatečné nastavení pro autentizaci. Aplikace umožňuje nastavit zabezpečení za využití normálních přihlašovacích údajů, TLS

autentizace (privátní a veřejný klíče), OAuth (speciální typ autentizace) a speciálních certifikátů.

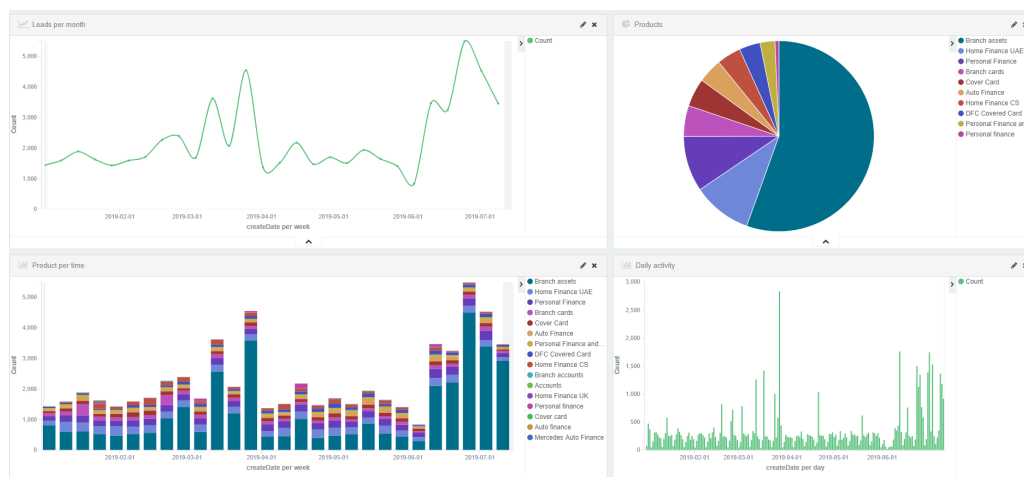
Dále je nutné nastavit jméno indexu, jenž má být aplikací využíván. Jelikož Grafana automaticky filtruje data podle času přidání, je nutné vyplnit název pole obsahující informaci o čase. V případě Elasticsearch je dané pole většinou nazváno timestamp. Na index je možné aplikovat časové vzorky, jež jsou vytvářené některými aplikacemi (například Logstash běžně vytváří slovníky podle času).

Jak již bylo dříve zmíněno, aplikace vyžaduje nastavení využívané verze nástroje Elasticsearch, verze bohužel není automaticky detekována. Pokud je nastavena špatná verze a data ze serveru neodpovídají předpokládanému formátu, aplikace později nemusí fungovat správně.

Po vyplnění všech požadovaných polí je možné nastavení uložit, čímž je automaticky i otestováno spojení se serverem. V případě neúspěšného spojení se v dolní části okna objeví popisek s bližšími informacemi o chybě. Nastavení datového zdroje je možné kdykoliv změnit. Příklad vhodného připojení k Elasticsearch je vidět na obrázku 5.2.

5.2 Vlastní vizualizace

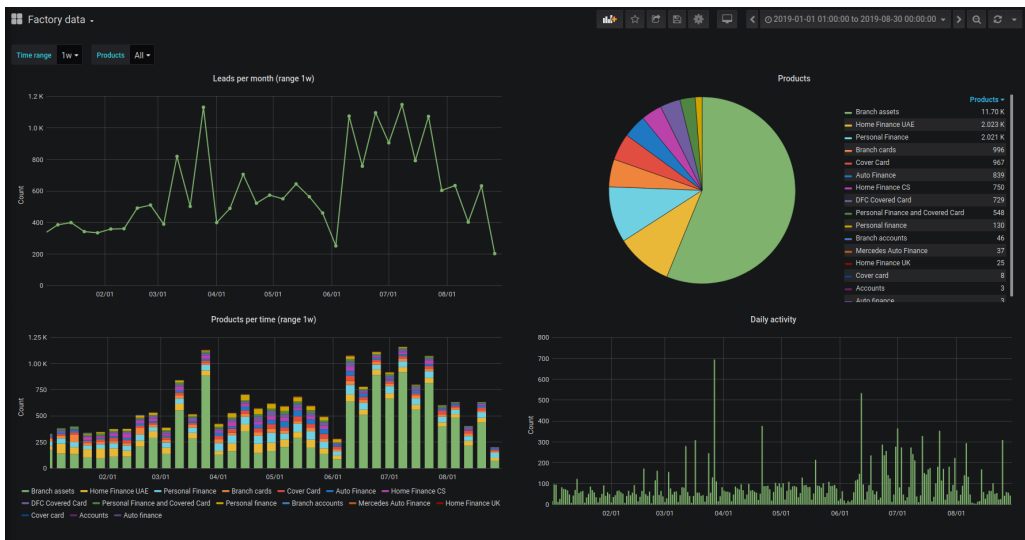
Dotázaná firma dodala mimo dat a konfiguračních souborů i jimi vytvořený dashboard (dashoard aplikace Kibana) jako příklad požadovaného výsledku. Vizualizace vytvořená za účelem této práce má tudíž mít stejnou informační hodnotu jako poskytnutý vzor, jež je možné vidět na obrázku 5.3.



Obrázek 5.3: Příklad požadovaného výstupu

Požadovaná vizualizace není příliš složitá, nejsložitějším prvkem je sklá-

daný sloupcový graf (graf v levém dolním rohu). Ale i tento prvek je možné v Grafaně vytvořit přidáním několika sérií dat do obvyklého sloupcového grafu. Jelikož neznáme počet sérií, bylo vhodné agregovat data podle specifického pole (v tomto případě pole produktů). Požadovaný dashboard obsahuje koláčový graf, tudíž bylo před samotnou vizualizací nutné nainstalovat příslušný plugin z webové stránky, jak bylo popsáno v části 3.2.1. Po instalování pluginu bylo již možné vytvořit výslednou vizualizaci, jež je možné vidět na obrázku 5.4.



Obrázek 5.4: Výsledná vizualizace

Po důkladnějším prozkoumání vzorové vizualizace bylo patrné, že mimo defaultního časového filtru bylo nutné vytvořit jeden dodatečný filtr pro produkty. Jak bylo rovněž popsáno v části 3.2.1, Grafana umožňuje pouze vytvoření dropdown filteru nad celou vizualizací. Jelikož bylo předem známé pole nad nímž bylo nutno filtrovat, před tvorbou samotné vizualizace byla vytvořena příslušná proměnná *products*.

K navýšení interaktivity grafů byla vytvořena časová proměnná *time*, tato proměnná byla dále využita k agregování dat u grafů v levé části. Jelikož pro malé časové intervaly není čárový graf příliš vhodný, byl u pravého horního grafu přepsán styl série podle proměnné *time*. Pro interval jedné hodiny je zobrazen bodový graf, zatímco pro denní interval jsou data zobrazena prostřednictvím sloupců. Tyto vlastnosti vytvořené vizualizace není možné reprezentovat obrázkem, pro bližší pochopení popsaných vlastností je k práci přiloženo ilustrativní video *dashboard.mp4*.

Vizualizace byly navrženy v tmavém motivu s základní barevnou paletu. Barevný model byl vybrán tak, aby byly grafy lehce čitelné a vizuálně pří-

jemné, nicméně uživatel si může spektrum přizpůsobit vlastním potřebám. U jednotlivých grafů byla nastaveno průhledné pozadí panelu.

5.3 Detekované nedostatky

Navržená vizualizace je plně funkční, informační hodnota je totožná s uživatelem poskytnutým vzorem. Aplikací nabízené filtry nejsou sice uživatelsky přívětivé jako vestavěné filtry aplikace Kibana, ale k ovládání grafů jsou dostačující. Výsledky není možné filtrovat dalšími dotazy nad daty, veškeré možné změny je možné provádět pouze zmíněným dropdown fitrem a vestavěným časovým filrem. Naopak existence proměnných umožňuje měnit styly jednotlivých graf v závislosti na použitých filtrech. Díky tomu je možné kombinovat několik typů grafu do jednoho panelu.

U vizualizací byly zachovány popisky z vzorové vizualizace (titulky grafů na levé straně byly rozšířeny o informaci o použitém časovém intervalu), tyto titulky není možné skrýt. Oproti původnímu návrhu firmy, titulky v aplikaci Grafana nejsou příliš výrazné. K ose x není možné přiřadit další popisky. Představuje-li osa x časovou osu, není ani možné určit po jakém intervalu budou jednotlivé úseky popsány.

Legendy grafů jsou zobrazovány pouze u grafů, kde je to nutné. Pokud je v grafu zobrazována pouze jedna série dat, legenda je skryta. U koláčového grafu bylo nastaveno umístění legendy napravo od vizualizací, zatímco u skládaného sloupcového grafu byla legenda umístěna pod graf. Pokud by bylo zákazníkem požadováno, je možné umístění legend sjednotit. U grafů je také možné legendu zobrazit formou tabulky.

Dashboard je možné sdílet přímým odkazem nebo odkazem na statický snapshot. Pokud je požadováno vložení do stránky jako iframe, je vhodné upravit konfigurační soubor Grafany a umožnit otevřít vizualizace bez nutnosti přihlášení. V tomto případě by měl být daný dashboard před sdílením uzamknut. V případě nechtěného znehodnocení vizualizací je ovšem vždy možné navrátit dashboard do předchozích verzí za pomoci verzovacího systému zmíněného v 3.2.1.

Nezávisle na možnosti sdílení je v aplikaci zachován boční ovládací panel. Pokud je vyžadováno sdílet vizualizace bez tohoto panelu, je možné sdílet takzvaný kiosk mód přidáním `?kiosk` za sdílené url. Tento mód bohužel skryje i filtrovací nástroje. Při vkládání vizualizace do stránky jako iframe může být boční panel překryt jiným elementem, ale z důvodu cross origin reference není možné element skrýt jiným způsobem.

6 Závěr

V zadání bakalářské práce byly definovány zásady pro zpracování, určující rozsah této studie. V první části práce byla představena technologie ElasticSearch spolu se zbylými nástroji Elastic Stack. Také bylo popsáno několik nástrojů schopných pracovat nad ElasticSearch serverem. Byly porovnány jejich schopnosti vizualizace, interaktivity a integrace do webového prostředí. Následně byl popsán formát dat a jejich začlenění do serveru. Vzhled požadované vizualizace byl analyzován a v rozsahu schopném aplikací Grafana realizován. V posledním bodu práce byly popsány nedostatky, jež byly ve vzniklé vizualizaci detekovány.

Literatura

- [1] *Cross-Origin Resource Sharing (CORS)* [online]. Mozilla and individual contributors, 2020. [cit. 2020/4/10]. Dostupné z: <https://developer.mozilla.org/en-US/docs/Web/HTTP/CORS>.
- [2] *Elastic Stack - Meet the core products* [online]. Elasticsearch B.V., 2019. [cit. 2019/11/30]. Dostupné z: <https://www.elastic.co/products/elastic-stack>.
- [3] *elasticdump* [online]. TaskRabbit, 2020. [cit. 2020/4/10]. Dostupné z: <https://npmjs.com/package/elasticdump>.
- [4] *ElasticSearch - The heart of the Elastic Stack* [online]. Elasticsearch B.V., 2019. [cit. 2019/11/30]. Dostupné z: <https://www.elastic.co/products/elasticsearch>.
- [5] *Grafana* [online]. Grafana Labs, 2019. [cit. 2019/11/30]. Dostupné z: <https://grafana.com/grafana/>.
- [6] *Grafana documentation* [online]. Grafana Labs, 2017. [cit. 2019/11/30]. Dostupné z: <https://grafana.com/docs/v4.3/>.
- [7] *Kibana - Your window into the Elastic Stack* [online]. Elasticsearch B.V., 2019. [cit. 2019/11/30]. Dostupné z: <https://www.elastic.co/products/kibana>.
- [8] *Kibana Guide* [online]. Elasticsearch B.V., 2019. [cit. 2019/11/30]. Dostupné z: <https://www.elastic.co/guide/en/kibana/current/index.html>.
- [9] *Overview / Knowi* [online]. Knowi, 2017. [cit. 2019/11/30]. Dostupné z: <https://docs.knowi.com/hc/en-us/articles/115003551067-Overview-Knowi>.
- [10] *Knowi University* [online]. Knowi, 2017. [cit. 2019/11/30]. Dostupné z: <https://docs.knowi.com/hc/en-us/sections/115000738914-University>.
- [11] MANNING, C. D. – RAGHAVAN, P. – SCHÜTZE, H. *Introduction to information retrieval*. New York: Cambridge University Press, 2008. ISBN 978-0-521-86571-5.
- [12] *Modern Analytic Apps for the Enterprise - Plotly* [online]. Plotly, 2020. [cit. 2020/3/7]. Dostupné z: <https://plot.ly/>.

- [13] *R - elastic tutorial* [online]. rOpenSci, 2019. [cit. 2019/11/30]. Dostupné z: https://ropensci.org/tutorials/elastic_tutorial/.
- [14] *Convert R Markdown to PDF or HTML* [online]. Earth Lab, 2019. [cit. 2019/11/30]. Dostupné z: <https://www.earthdatascience.org/courses/earth-analytics/document-your-science/knit-rmarkdown-document-to-pdf/>.
- [15] *What is R?* [online]. The R Foundation, 2019. [cit. 2019/11/30]. Dostupné z: <https://www.r-project.org/about.html>.
- [16] ROUSE, M. *RESTful API* [online]. WhatIs.com, 2015. [cit. 2019/6/19]. Dostupné z: <https://searcharchitecture.techtarget.com/definition/RESTful-API>.
- [17] URBANEK, S. *R - JDBC* [online]. 2019. [cit. 2019/11/30]. Dostupné z: <https://www.rdocumentation.org/packages/RJDBC/versions/0.2-7.1/topics/JDBC>.
- [18] *Vega-Lite – A Grammar of Interactive Graphics* [online]. Vega-Lite, 2019. [cit. 2019/11/30]. Dostupné z: <https://vega.github.io/vega-lite/>.