

Arhitekture sistema velikih skupova podataka

Uvod

Arhitekture sistema velikih skupova podataka, dr Vladimir Dimitrieski

1

Sadržaj

- Uvod
- Inženjering podataka i nauka o podacima
- Veliki skupovi podataka
- Arhitekture sistema velikih skupova podataka (ASVSP)
- Principi projektovanja ASVSP
- Primena i izazovi ASVSP
- Uloge u timu za rad sa podacima
- Sadržaj kursa
- Primeri projekata

2

2

1

Uvod

3

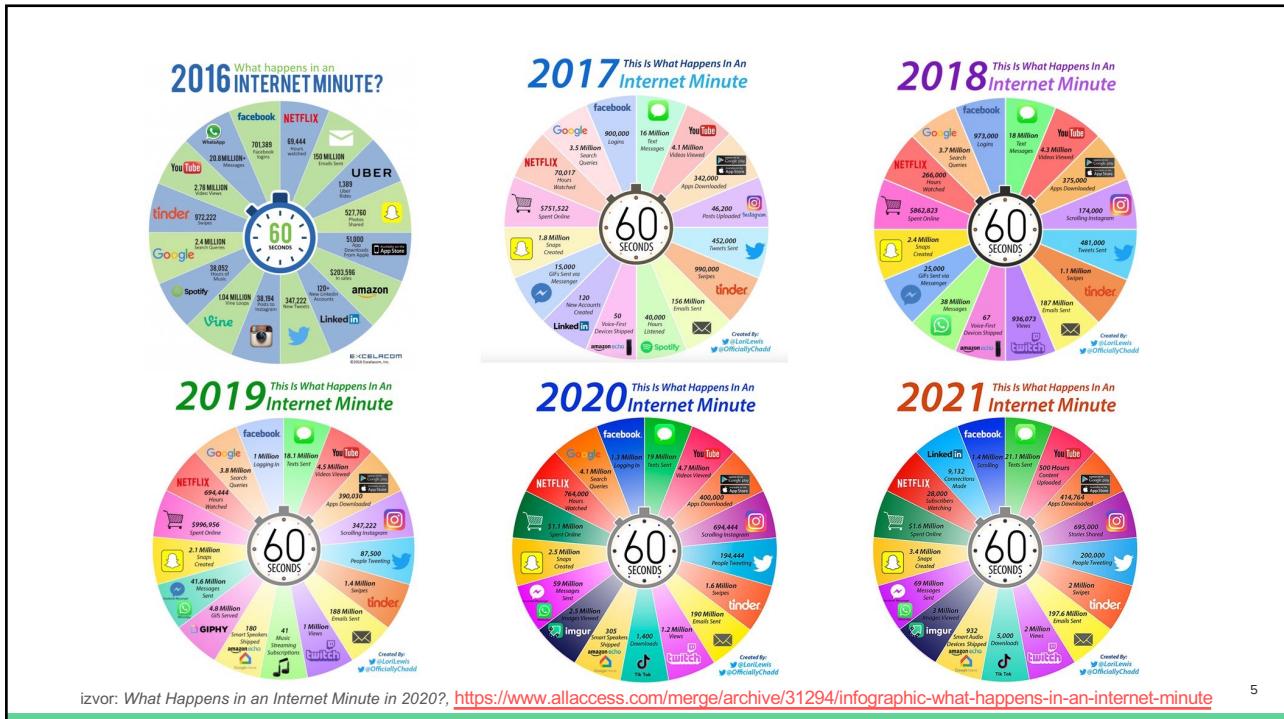
Uvod

- U poslednjih 15 godina dolazi do “eksplozije” podataka
 - veliki skupovi podataka su postojali od 60ih i 70ih godina 20og veka
 - rešenje problema upravljanja njima uključivalo je prelazak na relacione baze podataka
 - oko 2005. godine se uočava se **trend povećanog generisanja podataka** od strane ljudi
 - korišćenjem servisa poput Facebook-a i YouTube-a i sveprisutnih pametnih uređaja
 - uočavaju se ograničenja tradicionalnih tehnika skladištenja i upravljanja podacima
 - prvenstveno relacionih baza podataka
 - ograničenja se odnose na mogućnost skaliranja i performansi
 - nakon 2005. godine svakodnevno se povećava količina generisanih podataka
 - trenutno se generiše više od **30.000GB svake sekunde**
 - sa tendencijom da se taj broj drastično poveća
 - jer osim ljudi podatke sada generiše i sve veći broj IoT uređaja
 - mnogo izvora podataka i različitih oblika/formata podataka

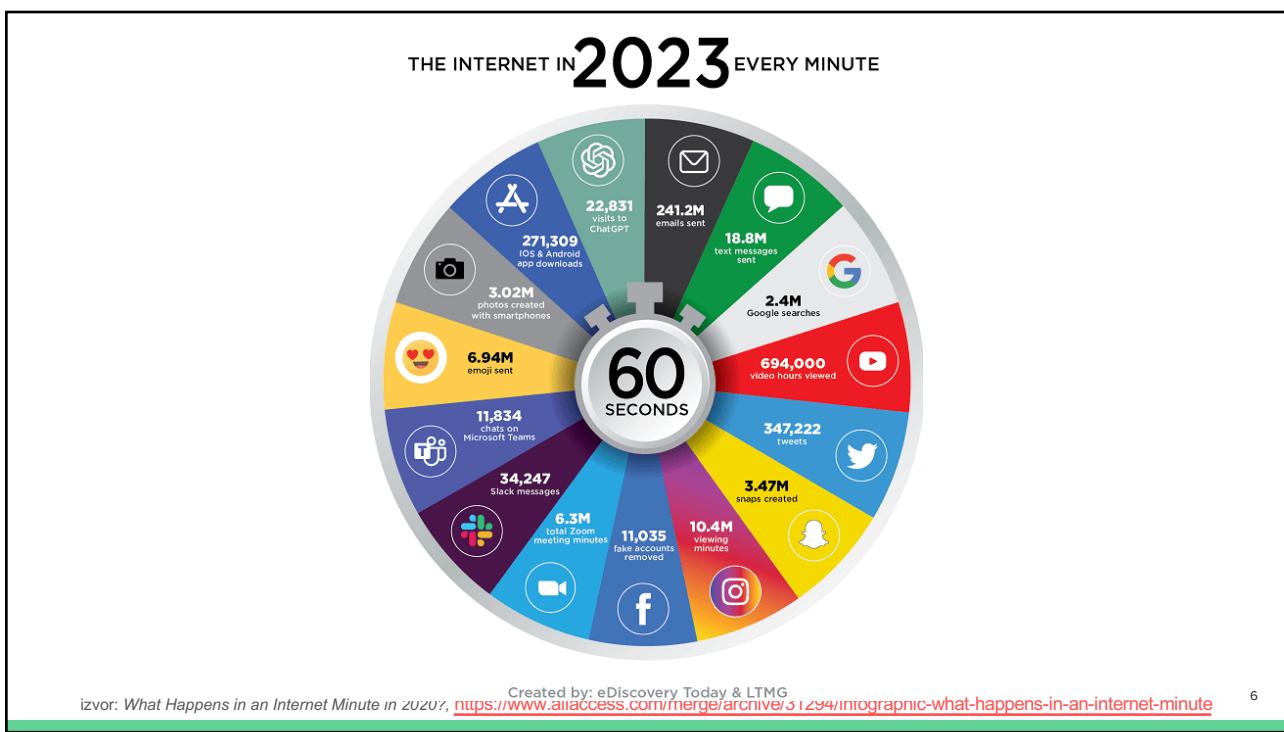
4

4

2



5



6

Uvod

- “**Data is the new oil.**” — Clive Humby (UK data scientist)
- “Data really powers everything that we do.” — Jeff Weiner (LinkedIn ex-CEO)
- “Without big data, you are blind and deaf and in the middle of a freeway.” — Geoffrey Moore (US business expert)
- “You can have data without information, but you cannot have information without data.” — Daniel Keys Moran (US programmer and author)
- “**Big data is at the foundation of all of the megatrends that are happening today, from social to mobile to the cloud to gaming.**” — Chris Lynch (US IT entrepreneur)

7

7

Uvod

- “With data collection, ‘the sooner the better’ is always the best answer.” — Marissa Mayer (Yahoo! ex-CEO)
- “**Without a systematic way to start and keep data clean, bad data will happen.**” — Donato Diorio (DtaZ CEO)
- “**Big data isn’t about bits, it’s about talent.**” — Douglas Merrill (US IT entrepreneur)

8

8

Uvod

- Nastaju nove naučne discipline i radna mesta
 - **inženjering podataka** (engl. *Data Engineering*)
 - **nauka o podacima** (engl. *Data Science*)
- Nastaju nove tehnologije i metodologije obrade velikih skupova podataka
 - pažnja se usmerila na **paralelizaciju** obrade i na **horizontalno skaliranje** sistema
 - uz težnju da se smanji kompleksnost obrade velikih količina podataka
 - mahom proistekle iz projekata velikih kompanija kao što su *Google* i *Amazon*
 - ali postoji i veliki broj projekata otvorenog koda
 - nastaju mnoge NoSQL baze podataka
 - sa različitim ciljevima i mogućnostima
 - nastaju algoritmi za paralelnu obradu podataka (npr. *MapReduce*)

9

Uvod

- Nastaju nove tehnologije i metodologije obrade velikih skupova podataka
 - teži se ka zameni postojećih sistema skladišta/baza podataka
 - glavni faktori koji utiču na nemogućnost rada sa velikim skupovima podataka
 - brzina traženja na disku
 - koja je niska
 - mogućnost skaliranja
 - koja nije dovoljno dobra

10

10

5

Brzina traženja na disku

- Brzina traženja na disku
 - ne razvija se istim tempom kao i veličina diska niti kao brzina prenosa podataka
 - obuhvata vreme potrebno da se glava diska pozicionira za čitanje sektora
 - kod tradicionalnih sistema, operacija traženja je dominantna
 - B-stablo je dobro za rad sa manjim delovima baze podataka
 - nepogodno sa ažuriranje ili čitanje većeg dela skladištenih podataka
 - drastično usporava rad sa velikim količinama podataka
 - koji se vrlo često čitaju redosledno i u celini

11

11

Skaliranje

- Skaliranje u tradicionalnim sistemima
 - jedan od osnovnih problema tradicionalnih sistema
 - u kojima se koriste relacione baze podataka
 - sa povećanjem količine podataka
 - više se brine o arhitekturi sistema
 - redovi čekanja, particije, replikacija itd.
 - nego o samoj poslovnoj logici
 - relacione baze podataka
 - nisu pogodne za horizontalno particonisanje
 - usled izuzetno loših performansi spojeva
 - mali broj relationalnih baza podataka podržava interno horizontalno particonisanje

12

12

Inkrementalne arhitekture

- Inkrementalne arhitekture
 - veoma rasprostranjene
 - koriste baze podataka za upis i ažuriranje podataka
 - održavajući ažurnu sliku stanja realnog sveta
 - direktnim ažuriranjem prethodne slike stanja
 - inkrementalno ažuriranje podataka
 - umesto čuvanja svih izvornih podataka čuvaju se agregati
 - npr. ukupan broj pristupa web stranici
 - umesto pojedinačnih pristupa
 - npr. ukupna plata zaposlenog nakon svih povišica
 - umesto osnovne plate i povišica odvojeno
 - ne zavise od izbora relacione ili NoSQL baze podataka
 - zavise od **pristupa projektovanju** šeme baze podataka
 - zavise od **odabira algoritama** za rad sa podacima



izvor: *Big Data: Principles and best practices of scalable real-time data systems*, Nathan Marz, James Warren

13

13

Inkrementalne arhitekture

- Inkrementalne arhitekture - Primer
 - aplikacija za praćenje analitike posete web prezentacijama
 - omogućava praćenje broja poseta definisanom URL-u
 - potrebno je da podrži izveštaj o 100 najposećenijih URL-ova
 - Šema relacione baze podataka
 - *id* - veštački ključ
 - *user_id* - identifikator korisnika sistema koji zahteva praćenje poseta određenoj stranici
 - *url* - adresa stranice za koju se prati broj poseta
 - *pageviews* - ukupan broj pristupa stranici

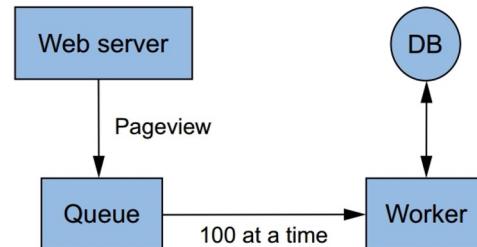
Column name	Type
<i>id</i>	integer
<i>user_id</i>	integer
<i>url</i>	varchar(255)
<i>pageviews</i>	bigint

izvor: *Big Data: Principles and best practices of scalable real-time data systems*, Nathan Marz, James Warren ¹⁴

14

Inkrementalne arhitekture

- Inkrementalne arhitekture - Primer
 - **Problem 1:** porast broja korisnika dovodi do problema sa uvećavanjem brojača
 - Greška: engl. *Timeout error on inserting to the database*
 - prouzrokovana nemogućnošću baze podataka da isprati broj poseta i ažuriranja
 - **Rešenje 1:** umesto pojedinačnog uraditi paketna ažuriranja
 - uvesti red čekanja između aplikativnog servera i baze podataka
 - periodično preuzimati podatke iz reda i raditi paketno (objedinjeno) ažuriranje BP
 - obrada u radnom čvoru (engl. *worker*)
 - npr. na svakih 100 elemenata u redu



izvor: *Big Data: Principles and best practices of scalable real-time data systems*, Nathan Marz, James Warren 15

15

Inkrementalne arhitekture

- Inkrementalne arhitekture - Primer
 - **Problem 2:** prepunjava se red usled prevelikog broja zahteva
 - **Rešenje 2.1:** povećati broj radnih čvorova
 - ponovo dolazi do zagušenja BP usled prevelikog broja ažuriranja
 - vraćamo se na problem 1
 - **Rešenje 2.2:** povećati broj radnih čvorova i horizontalno particionisati BP (engl. *sharding*)
 - svaka particija sadrži deo sadržaja tabele
 - postoji direktna veza između particije i ključa torke
 - npr. pronalazi se hash ključa i smešta se u particiju čija je adresa dobije kao ostatak pri deljenju hash-a sa brojem particija
 - uniformna raspoređenost vrednosti ključeva po particijama
 - aplikativni server može biti svestan particionisanja ili prepustiti upravljanje SUBP-u
 - postojanje particija dovodi do povećanja kompleksnosti upita
 - kako pronaći 100 najposećenijih URL-ova?

16

16

Inkrementalne arhitekture

- Inkrementalne arhitekture - Primer
 - **Problem 3:** dolazi do zagušenja na particijama usled prevelikog broja zahteva
 - **Rešenje 3:** povećati broj particija
 - svako povećanje broja particija zahteva ponovno reparticionisanje podataka
 - usled potrebe za uniformnom obradom podataka
 - potrebno obezbediti što manji gubitak novih podataka
 - nastao zbog nedostupnosti servera usled reparticionisanja baze podataka

17

17

Inkrementalne arhitekture

- Inkrementalne arhitekture - Primer
 - **Problem 4:** hardver na kojima se nalaze particije otkazuje
 - engl. *fault-tolerance issues*
 - što je veći broj particija to je veća verovatnoća da će doći do otkaza
 - **Rešenje 4:** uvesti replikaciju i redundantne redove
 - radni čvorovi pišu u redundantne redove
 - dok se ne oporavi particija
 - svaka particija se replicira na odgovarajući broj drugih particija
 - replikacione kopije služe isključivo za čitanje
 - ***više vremena se brinemo o arhitekturi sistema nego o samoj poslovnoj logici!***

18

18

Inkrementalne arhitekture

- Inkrementalne arhitekture - Primer
 - **Problem 5:** slučajno propagiranje i akumulacija greške
 - engl. *corruption issues*
 - slučajno uvedena greška u algoritam brojanja, koja uveća brojač za 2 umesto za 1
 - nije moguće smanjiti brojače poseta
 - **Rešenje 5:** izmena šeme baze podataka???
 - odbacivanje inkrementalnog pristupa obradi podatka

19

19

Inkrementalne arhitekture

- Inkrementalne arhitekture - izazovi i problemi
 - **operaciona složenost**
 - prouzrokovana složenošću upravljanja infrastrukturom nad kojom se izvršava aplikacija
 - npr. sažimanje indeksa u bazama podataka
 - intenzivna operacija koja zahteva dosta procesorskog vremena
 - zahteva izuzetno pažljivo podešavanje baze podataka
 - i planiranje u ranim fazama projektovanja sistema
 - može da prouzrokuje zagušenje sistema pa čak i otkaz
 - ako nije konfigurisana na odgovarajući način

20

20

10

Inkrementalne arhitekture

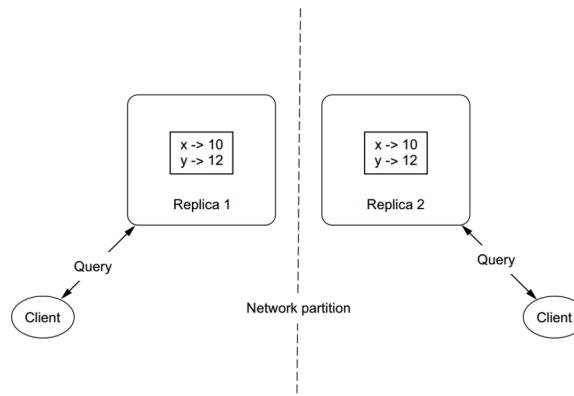
- Inkrementalne arhitekture - izazovi i problemi
 - **konvergentna konzistentnost**
 - **CAP teorema** za distribuirane sisteme
 - nije moguće istovremeno imati konzistentan sistem, dostupan sistem i sistem koji toleriše razdvojenost svojih delova
 - **žrtvuje se konzistentnost**
 - dozvoljeno je da u pojedinim trenucima sistem bude nekonzistentan
 - ali nakon razdvojenosti sistema da se vrati u konzistentno stanje
 - kako bi se izbeglo gubljenje podataka
 - zahteva pažljivo projektovanje šeme baze podataka i pristupa skladištenju podataka
 - zahteva pažljivu koordinaciju aplikacija i replikacije podataka

21

21

Inkrementalne arhitekture

- Inkrementalne arhitekture - izazovi i problemi
 - **konvergentna konzistentnost - primer**



izvor: *Big Data: Principles and best practices of scalable real-time data systems*, Nathan Marz, James Warren

22

22

Inkrementalne arhitekture

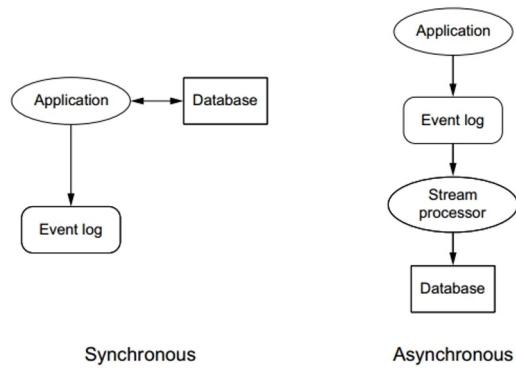
- Inkrementalne arhitekture - izazovi i problemi
 - nedovoljna otpornost na ljudske greške
 - ažuriranje stanja u bazi podataka se obavlja čak i u prisustvu ljudske greške
 - često nije moguće oporaviti se od greške
 - zbog kumulativne/agregatne osobine izmenjenog podatka
 - može se rešiti bez kompletne izmene arhitekture
 - dovoljno je uočiti problem prilikom projektovanja arhitekture i šeme baze podataka
 - umesto agregata čuvati pojedinačne zapise o svakom događaju iz realnog sveta

23

23

Inkrementalne arhitekture

- Inkrementalne arhitekture - izazovi i problemi
 - nedovoljna otpornost na ljudske greške - rešenje



izvor: *Big Data: Principles and best practices of scalable real-time data systems*, Nathan Marz, James Warren

24

24

Inkrementalne arhitekture

- Rešavanje problema u tradicionalnim sistemima za upravljanje podacima
 - potrebno pronaći **nove načine za upravljanje podacima**
 - potrebno pronaći **nove načine za unapređenje obrade podataka**
 - vertikalno ili horizontalno skaliranje
 - povećanje procesne moći
 - povećanje prostora skladištenja
 - ne postoji jedinstveno rešenje
 - različiti problemi zahtevaju različite pristupe
 - postoje tehnologije i arhitekturini principi koji mogu doprineti lakšem upravljanju velikom količinom podataka
 - izvršiti obuku novih generacija inženjera
 - u oblasti **inženjeringu podataka i nauke o podacima**
 - da bi mogli da rade u uslovima sve veće količine podataka

25

25

Inženjering podataka i nauka o podacima

26

13

Informacioni sistemi

- **Informacioni sistem (IS)** (engl. *Information system*)
 - model dela ili celog realnog sistema razvijen sa ciljem pružanja informacija, neophodnih za funkcionisanje i upravljanje sistemom
 - integrisani skup alata za sakupljanje, čuvanje, obradu i prenošenje **informacija**
 - u cilju donošenja boljih poslovnih odluka, koordinaciju, analizu i kontrolu u okviru organizacije
 - kombinacija hardvera, softvera i telekomunikacionih mreža
 - koju ljudi koriste kako bi sakupili, kreirali i distribuirali korisne informacije u okviru neke organizacije
- Mesto IS u sistemu
 - **infrastrukturna komponenta sistema**
 - deo upravljačkog sistema za sistem

27

27

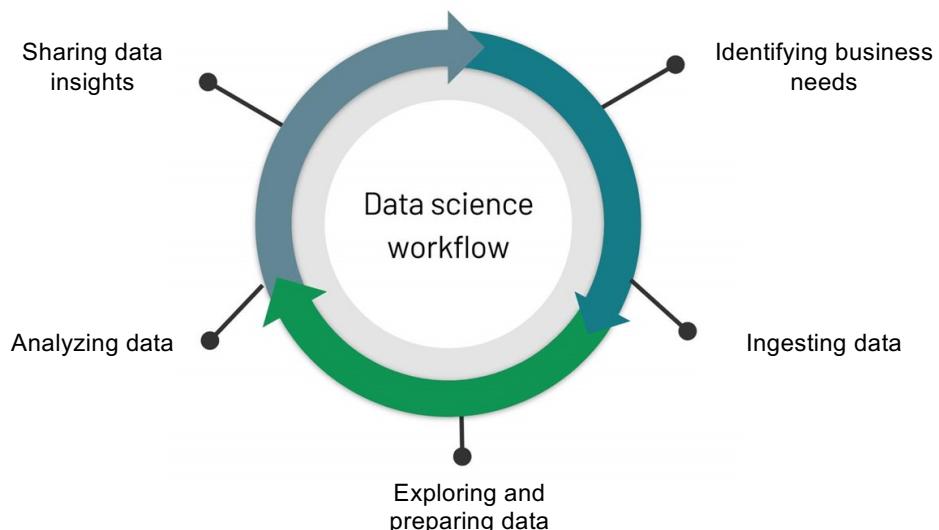
Nauka o podacima

- **Nauka o podacima** (engl. *Data science*)
 - skup fundamentalnih principa koji nas vode ka izdvajajuju informacija i znanja iz podataka
 - **interdisciplinarna oblast** koja obuhvata naučne metode, procese i sisteme
 - za izvlačenje znanja ili dobijanja uvida u značenje podataka
 - strukturiranih ili nestrukturiranih
 - objedinjuje discipline matematike, statistike, analize podataka i povezanih metoda
 - radi razumevanja i analiziranja realnih događaja i fenomena pomoću podataka

28

28

Proces rada u okviru nauke o podacima



izvor: Fundamentals of Big Data, Databricks <https://academy.databricks.com/>

29

29

Inženjering podataka

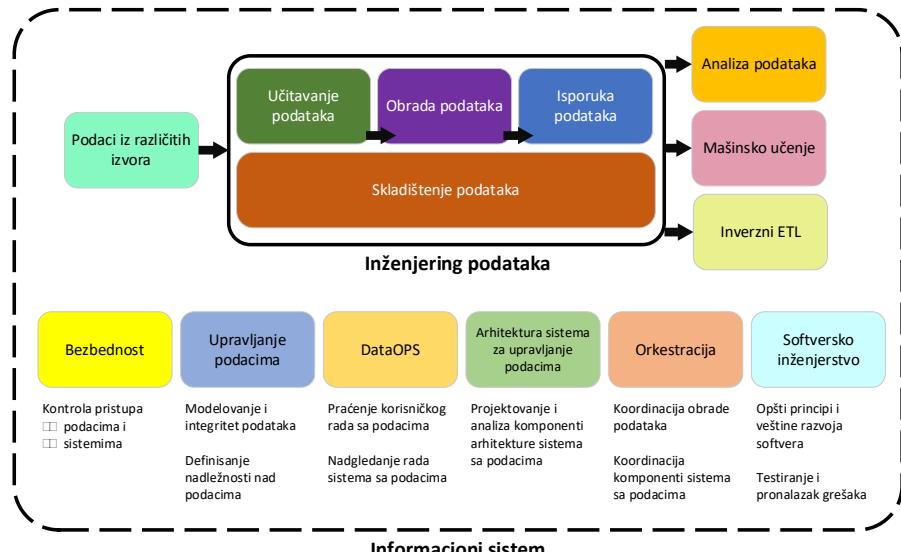
- **Inženjering podataka** (engl. *Data Engineering*)
 - aspekt nauke o podacima, **usmeren ka pozadinskim (engl. Backend)** zadacima u cilju obuhvata, obrade i isporuke podataka
 - obuhvat podataka i priprema za analizu
 - tokovi i pristup informacijama
 - obrade i podaci smaknuti u vremenu (engl. *data pipeline*)
 - iskoršćenje izvora velikih skupova podataka radi analiza
 - **primena velikih skupova podataka u realnom svetu**
 - ne bavi se krajnjom analizom podataka – izveštavajućom funkcijom IS
 - bavi se
 - arhitekturama sistema velikih skupova podataka
 - projektovanjem i konfiguracijom baza podataka
 - interfejsima ka izvorima podataka
 - tehnikama programiranja



30

30

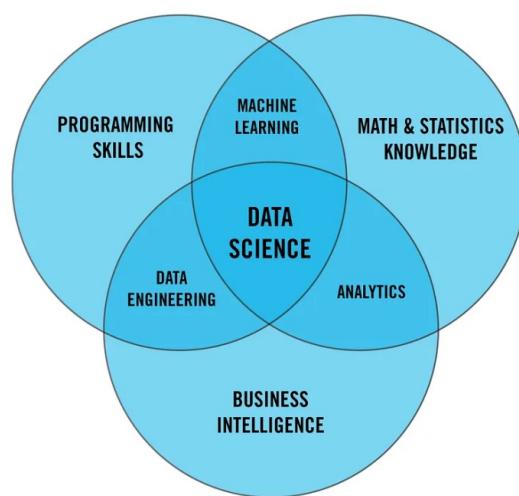
Inženjering podataka – povezani elementi



31

31

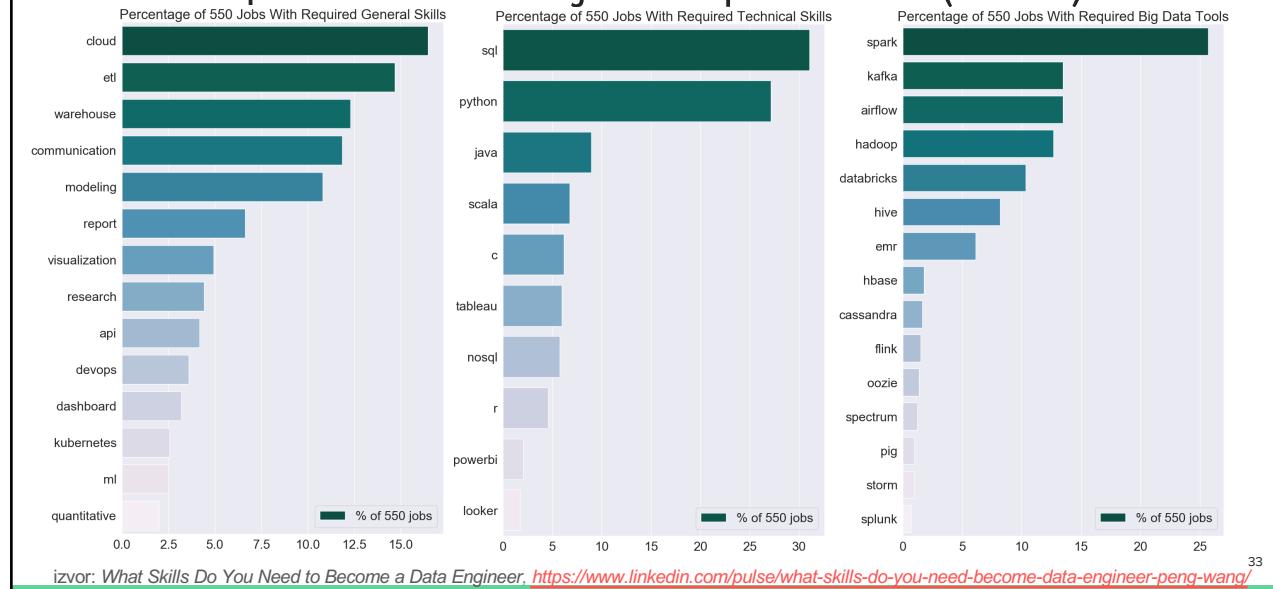
Nauka o podacima i inženjering podataka



32

32

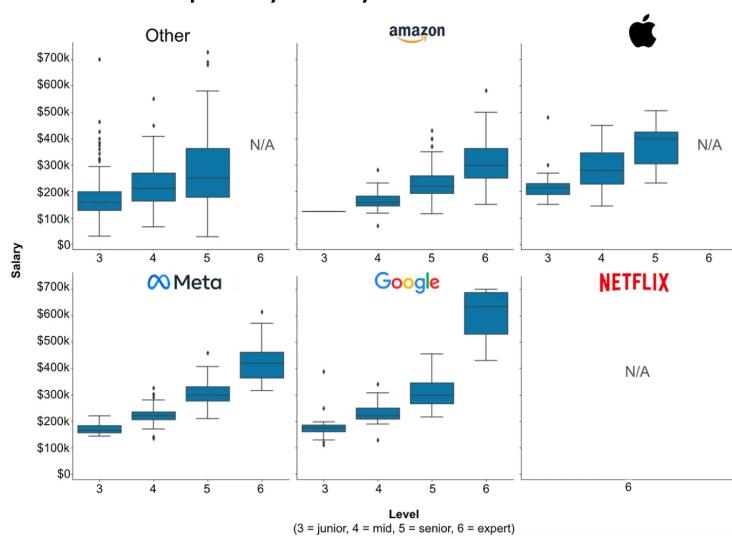
Veštine potrebne inženjerima podataka (2022)



33

Plate u oblasti inženjeringu podataka

2022 data salaries at FAANG companies sampled from more than 4000 data points by seniority



izvor: Data salaries at FAANG companies in 2022 <https://medium.com/@mikldd/data-salaries-at-faang-companies-in-2022-29d5b56b2428>

34

34

Veliki skupovi podataka

35

Veliki skupovi podataka

- **Veliki skup podataka**
 - engl. ***big data***
 - šta je veliki skup podataka?
 - Šta veliki skup podataka čini velikim?
 - ne postoji jasna i opšte prihvaćena definicija
 - različite kompanije imaju različito poimanje pojma **veliki skup podataka**
 - istorijski prva i najčešće korišćena/citirana definicija koju je dao *Doug Laney*
 - često se naziva i 3 V definicija
 - **veliki skup podataka** je **skup podataka** koji sadrži raznovrsne podatke (engl. *variety*) koji se prikupljaju u sve većem obimu (engl. *volume*) i sve većom brzinom (engl. *velocity*).

36

36

18

Veliki skupovi podataka

- Veliki skup podataka
 - veliki skup podataka je definisan kroz tri V osobine
 - **količina podataka** (engl. *volume*) - količina prikupljenih podataka koju treba obraditi
 - najčešće je potrebno obraditi nestrukturirane podatke
 - u zavisnosti od kompanije, meri se u stotinama GB pa čak i u desetinama ZB



37

izvor: Fundamentals of Big Data, Databricks <https://academy.databricks.com/>

37

Veliki skupovi podataka

- Veliki skup podataka
 - veliki skup podataka je definisan kroz tri V osobine
 - **brzina prikupljanja podataka** (engl. *velocity*) - količina podataka koji su prikupljeni u jedinici vremena
 - nekada se posmatra i kao brzina po kojoj se prikupljaju i analiziraju podaci
 - najbrže prikupljanje podataka je ono koje smešta podatke u radnu memoriju



38

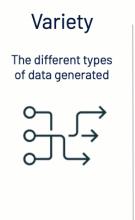
izvor: Fundamentals of Big Data, Databricks <https://academy.databricks.com/>

38

19

Veliki skupovi podataka

- Veliki skup podataka
 - veliki skup podataka je definisan kroz tri V osobine
 - **raznolikost podataka** (engl. *variety*) - dostupnost različitih tipova podataka u skupu
 - tradicionalni skupovi podataka su bili strukturirani sa jasno definisanim tipovima
 - danas, podaci se često prikupljaju u nestrukturiranom ili polustrukturiranom obliku



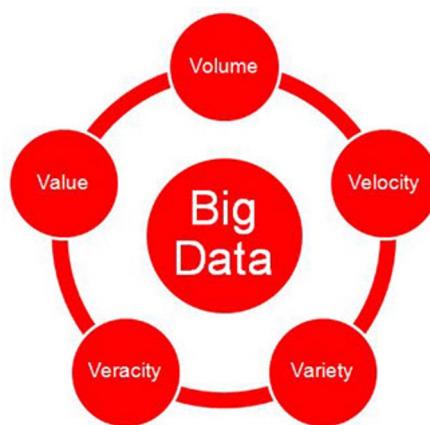
izvor: Fundamentals of Big Data, Databricks <https://academy.databricks.com/>

39

39

Veliki skupovi podataka

- Veliki skup podataka - pet V osobina



izvor: The 5 Vs of Big Data, Oracle, <https://blogs.oracle.com/step/the-5-vs-of-big-data-and-fujitsu-m10>

40

40

20

Veliki skupovi podataka

- Veliki skup podataka - dodatne osobine
 - **vrednost podataka** (engl. *value*) - vrednost koja može biti dobijena kao rezultat analize sprovedene nad prikupljenim podacima
 - karakteristika bitna za menadžment kompanije
 - vrednost koja im se može vratiti ulaganjem u prikupljanje i obradu velikih količina podataka
 - **verodostojnost podataka** (engl. *veracity*) - stepen istinitosti i tačnosti podataka
 - skup podataka može da sadrži i pogrešno/delimično prikupljene podatke, maliciozne podatke itd.
 - definiše da li su prikupljeni podaci odgovarajući za problem koji se rešava
 - od svih V osobina, **najteže je obezrediti verodostojnost podataka**

41

41

Veliki skupovi podataka

- Veliki skup podataka - alternativne definicije
 - **veliki skup podataka je skup podataka koji je prevelik za skladištenje u tradicionalnim bazama podataka ili previše složen za obradu u tradicionalnim informacionim sistemima**
 - različite kompanije imaju različite informacione sisteme
 - drugačija granica kada skup podataka postaje prevelik i previše kompleksan
 - 100 GB ~ 100 TB ~ 100 PB ~ 10 ZB
 - **veliki skup podataka je skup strukturiranih i nestrukturiranih podataka koji dozvoljava kompaniji da putem naprednih tehnika analize podataka izvuče informacije od vrednosti za njeno posovanje**
 - novija definicija usled razvoja IS-ova i BP-a
 - inherentno obuhvata i velike količine podataka
 - iako veličina nije ključna osobina već dobijena vrednost iz podataka

42

42

Veliki skupovi podataka

- Veliki skup podataka
 - raznolikost podataka
 - izuzetno je bitna jer utiče na **način na koji se obrađuju podaci**
 - tipovi podataka prema strukturi
 - **strukturirani podaci**
 - **nestrukturirani podaci**
 - **polustrukturirani podaci**

43

43

Veliki skupovi podataka

- Tipovi podataka prema strukturi
 - **strukturirani podaci**
 - postoji **šema** koja definiše format podataka
 - striktno zadovoljavaju predefinisani format
 - u praksi, očekuje se da takva šema bude razrađena do potrebnog stepena detaljnosti
 - uobičajeno **pogodni za analizu**
 - primer
 - podaci u relacionoj bazi podataka
 - sve torke imaju isti format

44

44

Veliki skupovi podataka

Structured

Conforms to a schema



Order	CustID	Month	Item	Color	Price
101	20051	Dec	Pen	Red	2.99
102	20045	Mar	Pencil	Blue Yellow Red	3.99
103	29584	May	Eraser	Blue	1.25
104	29584	May	Pen	White	2.25
105	29584	May	Pencil	Blue Yellow Red	2.99
106	27485	Jan	Eraser	Blue Yellow	2.75
107	29574	Jan	Marker	Green	1.75
108	24447	Feb	Marker	Yellow Blue	7.25
109	26466	Jul	Pen	Black Red	5.25
110	27467	Jun	Pencil	Black	2.95

izvor: Fundamentals of Big Data, Databricks <https://academy.databricks.com/>

45

45

Veliki skupovi podataka

- Tipovi podataka prema strukturi
 - **nestrukturirani podaci**
 - (A) može postojati šema koja definiše format podataka
 - tipovi podataka su, međutim, često "preširoki"
 - nedovoljno detaljni sa stanovišta semantike podataka
 - (B) ne postoji ni šema koja definiše format podataka
 - ne postoji nikakav ugrađeni format kojeg podaci prate
 - moguće je da postoji nekakav eksterni format koji opisuje semantiku podataka, ali on ne pripada strukturi
 - **najrasprostranjeniji** tip podataka
 - po IDC, 90% svih podataka su nestrukturirani
 - primeri
 - objave sa socijalnih mreža ili email poruke
 - tekstualni ili multimedijalni dokumenti
 - podaci su nestrukturirani sa stanovišta sistema za njihovo skladištenje

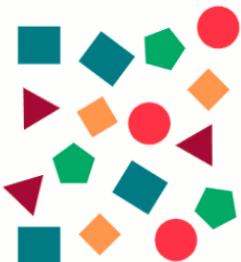
46

46

Veliki skupovi podataka

Unstructured

Does not fit neatly
Into a schema



47

izvor: *Fundamentals of Big Data, Databricks* <https://academy.databricks.com/>

47

Veliki skupovi podataka

- Tipovi podataka prema strukturi
 - **polustrukturirani podaci**
 - imaju definisanu strukturu ali postoje i odstupanja
 - atributi mogu da postoje
 - u svim entitetima
 - samo u nekim entitetima
 - prikazuju se, često, putem grafova i stabala
 - moguće je definisati šemu
 - koja specificira moguće elemente
 - koji učestvuju u strukturi
 - mogu ali ne moraju svi da postoje
 - primer
 - HTML kôd

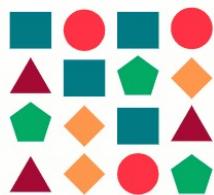
48

48

Veliki skupovi podataka

Semi-structured

Some level of organization



```

▼<div class="new-main-menu">
  ▼<div class="header-desktop-block">
    ▼<div class="container new-menu">
      ▶<a class="main-logo" rel="home" href="https://databricks.com/" title="Databricks">...</a>
      ▼<div id="new-m" class="menu-bar">
        ▼<div id="mega-menu-wrap-headerNew" class="mega-menu-wrap">
          ▶<div class="mega-menu-toggle">...</div>
          ▼<ul id="mega-menu-headerNew" class="mega-menu max-mega-menu mega-menu-horizontal" data-event="hover_intent" data-effect="fade_up" data-effect-speed="200" data-effect-mobile="disabled" data-effect-speed-mobile="0" data-panel-width="body" data-panel-inner-width="#new-m" data-mobile-force-width="false" data-second-click="close" data-document-click="collapse" data-vertical-behaviour="standard" data-breakpoint="1199" data-unbind="true">
            ▶<li class="mega-main-bar-li mega-menu-item mega-menu-item-type-custom mega-menu-item-object-custom mega-menu-item-has-children mega-menu-megamenu mega-
  
```

izvor: Fundamentals of Big Data, Databricks <https://academy.databricks.com/>

49

49

Veliki skupovi podataka

- Obrada podataka
 - aktivnost esencijalna za dobijanje vrednosti iz podataka
 - **podaci**
 - kolekcija znakova, cifara i simbola, bez predefinisanog značenja
 - potrebno je staviti podatke u kontekst radi dobijanja semantike
 - podatak = (entitet, obeležje, vreme, vrednost)
 - **informacije**
 - interpretirani podaci koji donose vrednost
 - obrađeni podaci kojima je pridodata značenje
 - obično nastaju kao posledica (automatizovane) obrade podataka
 - **znanje**
 - usvojene informacije koje se mogu koristiti za rešavanje problema
 - može biti eksplicitno i implicitno

50

50

Veliki skupovi podataka

- Obrada podataka
 - potrebno primeniti različite tehnike obrade nad podacima
 - radi dolaska do zaključaka i izvlačenja informacija i znanja iz podataka
 - najpopularnije tehnike koje se primenjuju
 - **veštačka inteligencija**
 - **mašinsko učenje**
 - **duboko učenje**
 - **statističke metode nauke o podacima**

51

51

Arhitekture sistema velikih skupova podataka
(ASVSP)

52

26

Arhitekture sistema velikih skupova podataka

- Arhitektura sistema velikih skupova podataka (ASVSP) - Definicija
 - engl. *big-data architecture*
 - **arhitektura sistema velikih skupova podataka je arhitektura sistema koja je projektovana tako da omogući obuhvat, obradu i analizu velikih skupova podataka**
 - **obuhvat** (engl. *ingestion*) - mogućnost prijema i skladištenja velikih količina podataka koji dolaze sa visokom učestanošću
 - **obrada** (engl. *processing*) - transformacija podataka u oblike pogodne za izvršenje upita i analizu podataka
 - **analiza** (engl. *analysis*) - rad sa podacima u cilju dobijanja informacije od vrednosti za osobu ili kompaniju koja je sprovodi

53

53

Arhitekture sistema velikih skupova podataka

- ASVSP - Poželjne karakteristike
 - **robustnost i tolerancija na otkaze**
 - engl. *robustness and fault tolerance*
 - samo otkaz svih komponenti sistema može dovesti do otkaza sistema kao celine
 - što viši stepen tolerancije na ljudske greške
 - **brzo čitanje i pisanje**
 - engl. *low latency reads and updates*
 - bez uticaja na robustnost sistema
 - **mogućnost skaliranja**
 - engl. *scalability*
 - obuhvata mogućnost očuvanja performansi u uslovima povećanja opterećenja
 - dodavanjem novih resursa

54

54

Arhitekture sistema velikih skupova podataka

- ASVSP - Poželjne karakteristike
 - **generalizacija**
 - engl. *generalization*
 - mogućnost pružanja podrške/usluga različitim aplikacijama
 - **proširivost**
 - engl. *extensibility*
 - dodavanje novih funkcionalnosti bez ili sa minimalnim narušavanjem postojeće arhitekture i minimalnom migracijom podataka
 - **podrška za *ad hoc* upite**
 - engl. *ad hoc queries*
 - usled potrebe za izvlačenjem nepredviđenih informacija iz postojećih podataka
 - potencijalno omogućava unapređenje poslovanja

55

55

Arhitekture sistema velikih skupova podataka

- ASVSP - Poželjne karakteristike
 - **lako održavanje**
 - engl. *minimal maintenance*
 - odabiranje komponenti sa što nižom implementacionom kompleksnošću
 - komponente koje su na pravom stepenu apstrakcije za domen u kojem se koriste
 - pažnja usmerena ka implementaciji poslovne logike
 - **mogućnost pronalaženja grešaka**
 - engl. *debuggability*
 - usled postojanja više komponenti, praćenje podataka od osnovnog oblika do njegovog krajnjeg oblika je esencijalno

56

56

Arhitekture sistema velikih skupova podataka

- Aspekti koji utiču na izbor elemenata ASVSP
 - **tip analize podataka** (engl. *analysis type*)
 - definiše da li je potrebna analiza podataka u realnom vremenu ili je potrebna paketna obrada podataka
 - utiče na izbor alata, gotovih rešenja, hardvera i izvora podataka
 - **metodologija obrade podataka** (engl. *processing methodology*)
 - definiše tip tehnike primenjene na obradu podataka
 - prediktivna analiza, *ad hoc* upiti ili izveštavanje
 - utiče na izbor alata i algoritama obrade podataka
 - **učestanost obuhvata podataka i veličina podataka** (engl. *data frequency and size*)
 - definiše koliko podataka se očekuje i kojom brzinom (količina/vreme) pristižu novi podaci
 - utiče na izbor mehanizma skladištenja i formata podataka kao i na izbor alata za preprocesiranje podataka

57

57

Arhitekture sistema velikih skupova podataka

- Aspekti koji utiču na izbor elemenata ASVSP
 - **tip podataka** (engl. *data type*)
 - semantički tip podataka koji će biti obradjeni
 - istorijski podaci, meta podaci, transakcioni podaci itd.
 - utiče na algoritme skladištenja i pisanja/čitanja podataka iz skladišta
 - **strukturiranost podataka** (engl. *content format*)
 - definiše nivo strukturiranosti podataka
 - strukturirani, nestrukturirani i polustrukturirani podaci.
 - utiče na izbor alata za procesiranje, prečišćavanje i skladištenje pristiglih podataka
 - **izvor podataka** (engl. *data source*)
 - definiše izvore iz kojih se podaci učitavaju
 - identifikacija svih potrebnih izvora podataka pomaže u definisanju opsega rešenja
 - sa tehnološke ali i poslovne tačke gledišta
 - utiče na izbor alata za procesiranje i prečišćavanje podataka

58

58

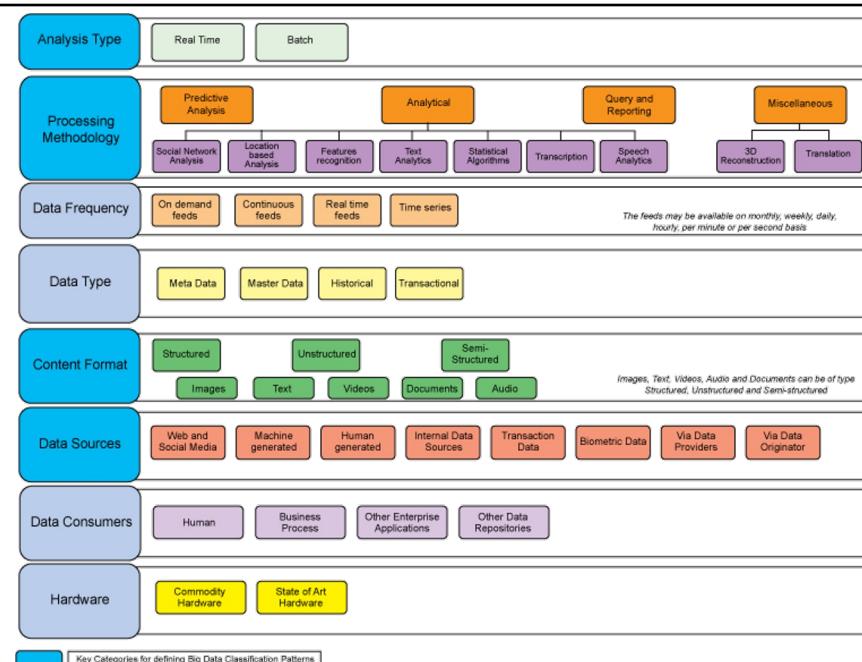
29

Arhitekture sistema velikih skupova podataka

- Aspekti koji utiču na izbor elemenata ASVSP
 - odredišta podataka** (engl. *data consumers*)
 - definišu entitete za koje su podaci prikupljani, obrađeni i kojima su podaci na kraju i prezentovani
 - npr. osobe ili poslovne aplikacije
 - utiče na izbor alata i prezentacionih formata i formi
 - hardver** (engl. *hardware*)
 - definiše tip hardvera na kojem će se svi podaci skladištiti i sve obrade podataka biti pokrenute
 - utiče na izbor alata i algoritama za obradu podataka

59

59



izvor: Big data architecture and patterns, IBM, <https://www.ibm.com/developerworks/analytics/library/bd-archpatterns2/index.html?ca=drs->

60

60

Arhitekture sistema velikih skupova podataka

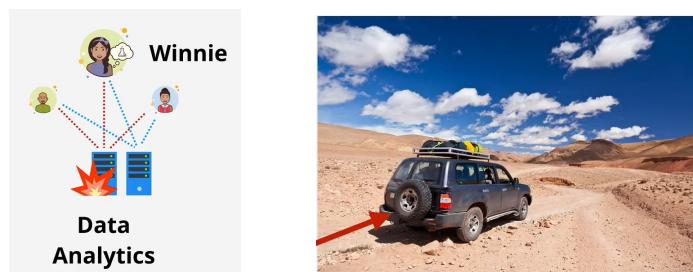
- Aspekti koji utiču na izbor elemenata ASVSP
 - **poslovni aspekti i korisnički zahtevi** (funkcionalni i nefunkcionalni)
 - ne samo da utiču na izbor komponenti arhitekture
 - već i na podešavanja i multiplicitet komponenti
 - **željeni nivo otpornosti na otkaze**
 - „**Sve će kad-tad otkazati!**“ (engl. “Everything fails“, Werner Vogels, CTO, AWS)
 - najviše uticaja na ASVSP imaju strategije za:
 - **visoku dostupnost** (engl. *high availability*)
 - **otpornost na otkaze** (engl. *fault tolerance*)
 - **oporavak od otkaza** (engl. *disaster recovery*)
 - **veoma bitno razlikovati visoku dostupnosti od otpornosti na otkaze**

61

61

Arhitekture sistema velikih skupova podataka

- **Visoka dostupnost**
 - osobina da je sistem **kontinualno dostupan** duži vremenski period
 - minimizacija nedostupnosti sistema
 - sistem može biti nedostupan jedan period vremena
 - **predviđeno ugovorom** sa pružaocem usluga
 - 99.9% = 8.77 sati godišnje dozvoljene nedostupnosti
 - 99.999% = 5.26 minuta godišnje dozvoljene nedostupnosti



izvor: AWS Architect Associate, Learn Cantrill, <http://learn.cantrill.io>

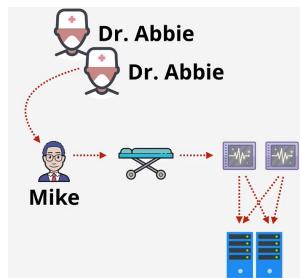
62

62

Arhitekture sistema velikih skupova podataka

- **Otpornost na otkaze**

- osobina sistema da nastavlja da neometano radi čak i u prisustvu delimičnog otkaza komponenti
- projektovanje da se potpuni **otkaz ne može dogoditi**
 - ili da je **verovatnoća otkaza minimalna**



izvor: AWS Architect Associate, Learn Cantrill, <http://learn.cantrill.io>

63

63

Arhitekture sistema velikih skupova podataka

- **Oporavak od otkaza**

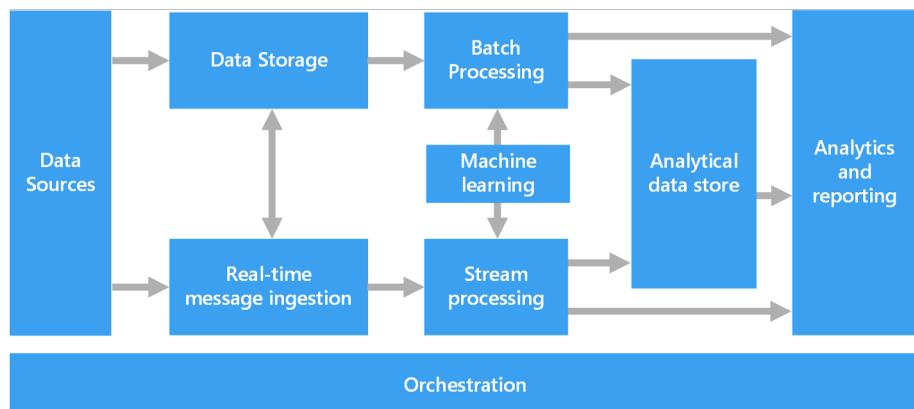
- obuhvata **skup polisa, alata i procedura** koje obezbeđuju **oporavak ili nastavak izvršavanja** esencijalnih infrastrukturnih elemenata ili sistema **nakon otkaza**
- izvršava se ukoliko osobine visoke dostupnost ili otpornosti na otkaze nisu bile ispunjene
- dve bitne karakteristike
 - u odnosu na koje se takođe vrši izbor i konfiguracija komponenti ASVSP
 - **ciljna tačka oporavka** (engl. recovery point objective, RPO)
 - količina podataka koja može biti izgubljena prilikom oporavka od otkaza
 - često merena kao jedinica vremena a ne količine podataka
 - **ciljno vreme oporavka** (engl. recovery time objective, RTO)
 - vreme za koje mora biti uspostavljen normalan rad sistema nakon otkaza
 - kako bi se izbegle neželjene posledice po poslovanje organizacije

64

64

Arhitekture sistema velikih skupova podataka

- Elementi ASVSP
 - uobičajni elementi arhitekture (ne moraju svi biti prisutni u svakoj arhitekturi)



izvor: *Big data architectures*, Microsoft, <https://docs.microsoft.com/en-us/azure/architecture/data-guide/big-data/>

65

Arhitekture sistema velikih skupova podataka

- Elementi ASVSP
 - **izvori podataka** (engl. *data sources*)
 - obuhvataju sve digitalne entitete iz kojih se mogu dobiti podaci
 - često generišu nestrukturirane ili polustrukturirane podatke
 - koji se u takvom obliku preuzimaju i skladište u sistemu
 - izuzetno bitno imati pouzdane izvore podataka
 - kako prikupljeni podaci ne bi doveli do pogrešnih rezultata prilikom analize
 - **utiče na verodostojnost podataka**
 - npr. baze podataka, datoteke različitih namena i uređaji koji šalju podatke u realnom vremenu

66

66

Arhitekture sistema velikih skupova podataka

- Elementi ASVSP
 - **skladišta podataka** (engl. *data storage*)
 - skladište velike količine podataka projektovano sa ciljem da podaci budu obrađeni u režimu paketne obrade
 - često su u pitanju distribuirani sistemi datoteka
 - optimizovani za horizontalno skaliranje
 - **paketna obrada podataka** (engl. *batch processing*)
 - obrada velike količine podataka radi transformacije podataka u oblik pogodan za analizu
 - obrađuju se paketi podataka a ne pojedinačni podaci
 - skladište mora biti optimizovano za čitanje i pisanje paketa podataka
 - naglasak je na paketnom čitanju
 - dugotrajna i paralelna obrada podataka
 - prisutno horizontalno skaliranje resursa

67

67

Arhitekture sistema velikih skupova podataka

- Elementi ASVSP
 - **obuhvat podataka u realnom vremenu** (engl. *real-time message ingestion*)
 - potreban u slučaju postojanja izvora podataka koji šalje podatke u realnom vremenu
 - npr. IoT uređaji
 - sve poruke dospele od izvora u realnom vremenu potrebno je prihvati i skladištiti
 - obuhvata redove čekanja i *buffer zone*
 - obično se zahteva da garantuje isporuku i obradu poruka
 - jer je ponovno generisanje i slanje poruke od izvora često nemoguće
 - **obrada tokova podataka** (engl. *stream processing*)
 - obrada podataka primljenih od izvora u realnom vremenu radi transformacije podataka u oblik pogodan za analizu
 - potrebno da obrada traje kraće nego što je slučaj sa paketnom obradom
 - jer se očekuje kraći odziv sistema i brže davanje odgovora na postavljene upite

68

68

Arhitekture sistema velikih skupova podataka

- Elementi ASVSP
 - **skladište podataka spremnih za analizu** (engl. *analytical data store*)
 - obuhvata podatke spremne za analizu i za postavljanje upita
 - rezultati paketne i obrade tokova podataka
 - često obuhvata strukturirane podatke
 - može da obuhvata relacione BP, relaciona skladišta podataka ali i NoSQL BP
 - **analiza podataka i izveštavanje** (engl. *analysis and reporting*)
 - cilj većine ASVSP je da pružaju usluge izveštavanja i analize velikih količina podataka
 - i kroz te aktivnosti da kreiraju vrednost i pruže nove informacije od interesa
 - **orkestracija** (engl. *orchestration*)
 - omogućava funkcionisanje ASVSP kao celine
 - osim velikog broja elemenata, svaki od elemenata može da bude implementiran kao distribuirani sistem na kojem se izvršava više instanci procesa

69

69

Arhitekture sistema velikih skupova podataka

- Tipični procesi podržani od strane ASVSP
 - **paketna obrada** velike količine uskladištenih podataka
 - **obrada podataka** koji se prikupljaju **u realnom vremenu**
 - **interaktivna pretraga** ("istraživanje") uskladištenih podataka
 - **prediktivna analiza** trendova u podacima i **podrška za mašinsko učenje**

70

70

Arhitekture sistema velikih skupova podataka

- **Paketna obrada podataka**

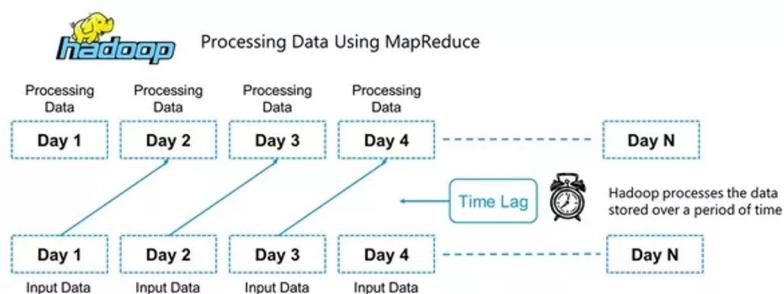
- efikasna obrada velike količine podataka
 - pristupa se svim podacima
 - obrađuje pakete podataka a ne pojedinačne podatke
- obrađuje podatke koji su prikupljeni duži vremenski period
 - npr. dani, meseci, godine
- obrada traje duži vremenski period
 - dovoljno dugo da se ova obrada ne bi posmatrala kao obrada u realnom vremenu
 - trajanje obrade se meri u minutima ili satima
- koriste alati/radni okviri kao što su *Hadoop* i *MapReduce*
- koristi se kada je potrebna dublja i detaljnija analiza podataka
 - od brze analize i dobijanja odgovora
- npr. obrada u sistemima za obračun plate i fakturisanje

71

71

Arhitekture sistema velikih skupova podataka

- Paketna obrada podataka



izvor: Big Data Battle : Batch Processing vs Stream Processing, Gowthamy Vaseekaran, <https://medium.com/@gowthamy/big-data-battle-batch-processing-vs-stream-processing-5d94600d8103>

72

72

Arhitekture sistema velikih skupova podataka

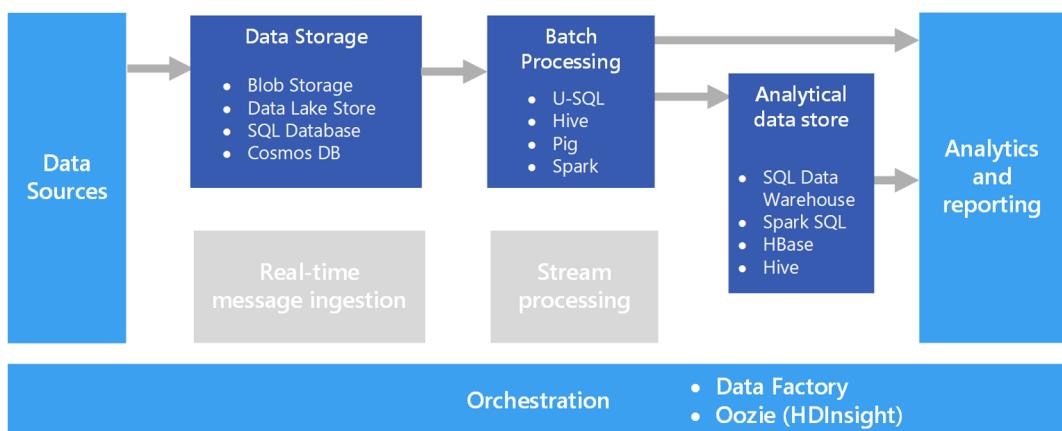
- Paketna obrada podataka
 - prednosti
 - pogodna i optimizovana za obradu velike količine podataka
 - visoka otpornost na uticaj ljudskih grešaka
 - obrada se može izvršavati neovisno od svakodnevnih aktivnosti sistema
 - kada je sistem slobodan ili manje zauzet
 - u slučaju korišćenja *cloud* platforme - kada je iznajmljivanje resursa povoljnije
 - održava trag obrade podataka
 - u svakom koraku obrade mogu se videti izvršene transformacije i oblik koji su podaci imali pre i nakon obrade
 - mane
 - kašnjenje između obuhvata novih podataka i dobijanja rezultata obrade
 - obrađeni podaci ne moraju biti najsvežiji u svakom trenutku
 - obrada podataka koju treba izvršiti samo jednom je obično veoma spora
 - jer usled prirode ovakve obrade često se ne piše na optimalan način

73

73

Arhitekture sistema velikih skupova podataka

- Paketna obrada podataka



izvor: Big data architectures, Microsoft, <https://docs.microsoft.com/en-us/azure/architecture/data-guide/big-data/batch-processing>

74

74

Arhitekture sistema velikih skupova podataka

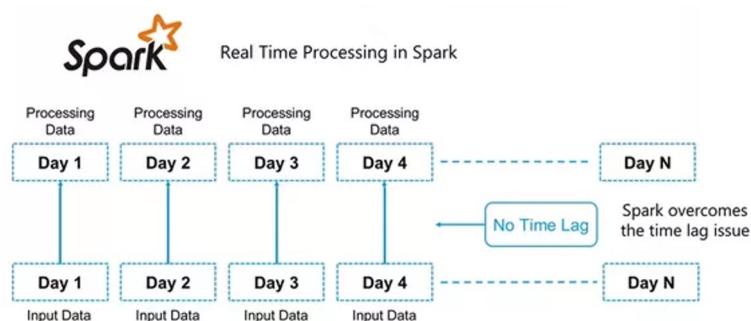
- **Obrada podataka u realnom vremenu**
 - obuhvata kontinualni obuhvat i obradu podataka
 - obrađuje pojedinačne podatke
 - može prikupljene podatke i da spoji u mikro-pakete i tako ubrza obradu
 - obrađuje podatke koji su prikupljeni kraći vremenski period
 - npr. sekunde ili minute
 - obrada traje kratko
 - mala količina podataka koja se obrađuje
 - obrada se događa pre snimanja pristiglih podataka na disk
 - koriste alati/radni okviri kao što su *Kafka* i *Storm*
 - koristi se kada je potrebna brza analiza i reakcija na neki događaj
 - npr. detekcija neovlašćenog pristupa ili detekcija pokušaja prevare ili napada
 - često se zapravo radi o obradi u skoro realnom vremenu (engl. *near-realtime data processing*)
 - jer postoji kašnjenje reda veličine nekoliko sekundi u dobijanju odgovora
 - npr. obrada u bankomatima i radarskim sistemima

75

75

Arhitekture sistema velikih skupova podataka

- Obrada podataka u realnom vremenu



izvor: Big Data Battle : Batch Processing vs Stream Processing, Gowthamy Vaseekaran, <https://medium.com/@gowthamy/big-data-battle-batch-processing-vs-stream-processing-5d94600d8103>

76

76

Arhitekture sistema velikih skupova podataka

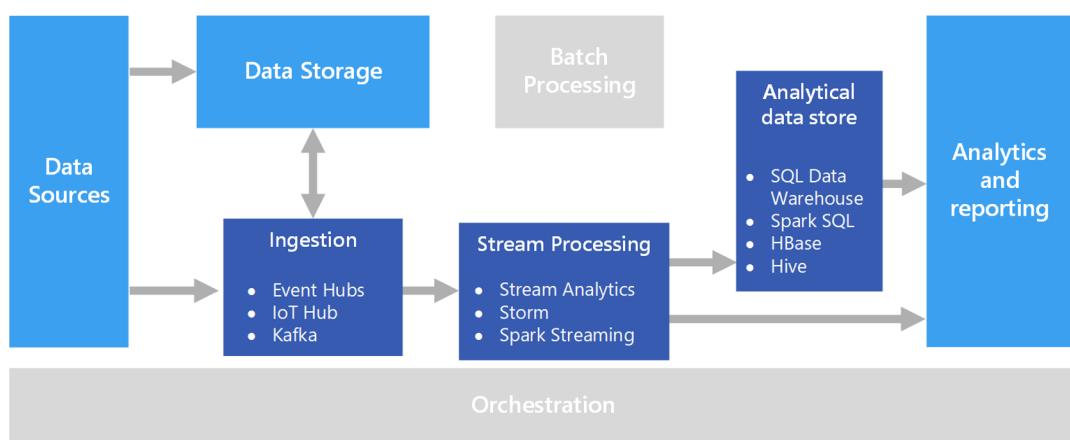
- Obrada podataka u realnom vremenu
 - prednosti
 - nema bitnijeg kašnjenja obrade podataka u odnosu na obuhvat podataka
 - obrađeni podaci predstavljaju najsvežiju sliku stanja sistema
 - jer se tek pristigli podaci skoro pa trenutno obrađuju
 - omogućava kompanijama i pojedincima da brzo reaguju na novootkrivene informacije
 - mane
 - algoritmi obrade podataka u realnom vremenu su po prirodi dosta kompleksni
 - kompleksno održavanje traga obrade podataka
 - manja otpornost na ljudske greške

77

77

Arhitekture sistema velikih skupova podataka

- Obrada podataka u realnom vremenu



78

izvor: Big data architectures, Microsoft, <https://docs.microsoft.com/en-us/azure/architecture/data-guide/big-data/real-time-processing>

78

Arhitekture sistema velikih skupova podataka

- Skladištenje podataka
 - podaci koji se obrađuju moraju biti skladišteni
 - i to u distribuiranom skladištu podataka velikog kapaciteta
 - najčešći načini skladištenja podataka
 - **skladišta podataka** (engl. *Data Warehouse*)
 - **jezera podataka** (engl. *Data Lake*)
 - **platforme sa jezerom podataka** (engl. *Data Lake Platforms*)

79

79

Arhitekture sistema velikih skupova podataka

- Skladištenje podataka
 - **skladišta podataka**
 - nastala 1980-ih godina
 - **obično sadrže strukturirane i prečišćene podatke**
 - podaci su često organizovani pomoću **relacionog modela podataka**
 - prednosti
 - stabilni i dugo postoje
 - zbog podrške samo za strukturirane podatke, laka su za održavanje i vršenje upita
 - mane
 - teško skaliraju
 - gube na vrednosti usled podrške samo za strukturirane podatke
 - skupi za održavanje i korišćenje
 - zajedno se naplaćuju i procesna moć i skladišni prostor

80

80

Arhitekture sistema velikih skupova podataka

- Skladištenje podataka
 - **jezera podataka**
 - **sadrže sirove (engl. raw) podatke**
 - mogu da sadrže i **nestrukturirane podatke**
 - prednosti
 - podrška za različite tipove podataka
 - lako skaliranje dodavanjem novih čvorova
 - odvojeno zakupljivanje procesne moći i skladišnog prostora
 - mane
 - teža navigacija kroz podatke usled nestrukturiranosti
 - usled velike količine podataka i nestrukturiranosti brzina upita može biti spora
 - potrebno ručno podešavanje upita i izvršilaca kôda

81

81

Arhitekture sistema velikih skupova podataka

- Skladištenje podataka
 - **platforme sa jezerom podataka**
 - **jezero podataka kao servis**
 - upravljan od strane druge, spoljne organizacije
 - prednosti
 - sve prednosti jezera podataka
 - jednostavnije kreiranje, održavanje i upravljanje
 - mane
 - zaključavanje za organizaciju koja nudi uslugu
 - nemogućnost finog podešavanja jezera podataka

82

82

Arhitekture sistema velikih skupova podataka

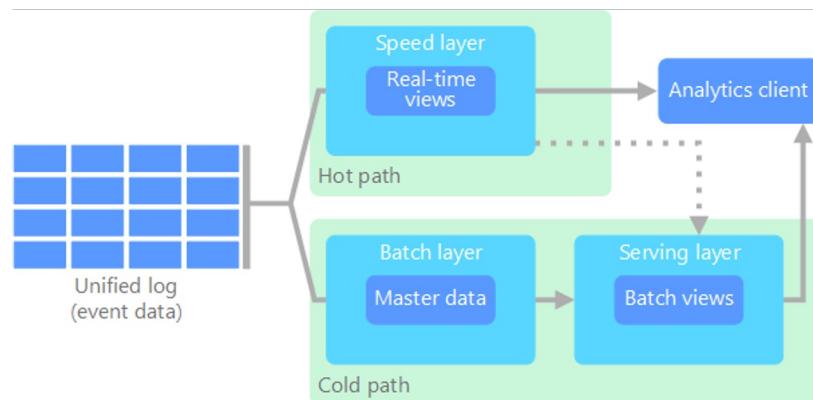
- ASVSP - Primeri arhitektura - Lambda
 - predlaže rešenje problema dugog čekanja na postavljeni upit kod sistema sa paketnom obradom podataka
 - podaci koji dođu u sistem, nikada se ne menjaju
 - samo se obrađuju i kreiraju se pogledi nad kojima se vrše upiti
 - paketna obrada
 - sa ponovnim računavanjem pogleda
 - traje dugo i izvršava se periodično u pozadini
 - obrada u realnom vremenu
 - samo nad podacima pristiglim od trenutka poslednjeg izračunavanja pogleda

83

83

Arhitekture sistema velikih skupova podataka

- ASVSP - Primeri arhitektura - Lambda



izvor: *Big data architectures*, Microsoft, <https://docs.microsoft.com/en-us/azure/architecture/data-guide/big-data/>

84

84

Arhitekture sistema velikih skupova podataka

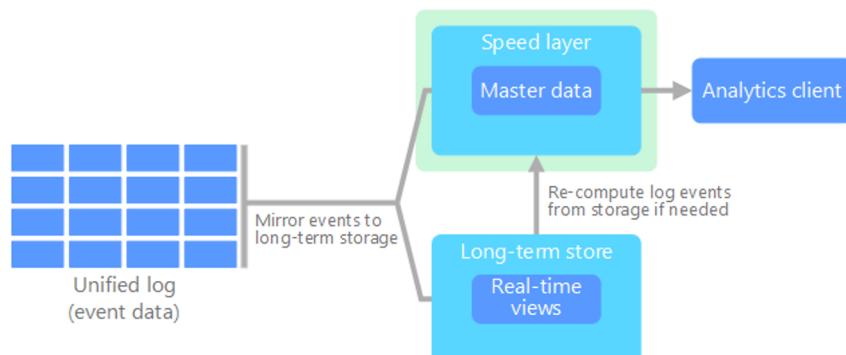
- ASVSP - Primeri arhitektura - Kappa
 - predlaže rešenje problema dvostrukе implementacije algoritama obrade arhitekture Lambda
 - podaci koji dođu u sistem, nikada se ne menjaju
 - samo se obrađuju i kreiraju se pogledi nad kojima se vrše upiti
 - svi podaci se šalju preko tokova podataka
 - jedna putanja podataka od izvora do konačnog obrađivača
 - moguće i ponovno sračunavanje svih pogleda
 - ponavljanjem toka podataka od početka

85

85

Arhitekture sistema velikih skupova podataka

- ASVSP - Primeri arhitektura - Kappa



izvor: *Big data architectures*, Microsoft, <https://docs.microsoft.com/en-us/azure/architecture/data-guide/big-data/>

86

86

Arhitekture sistema velikih skupova podataka

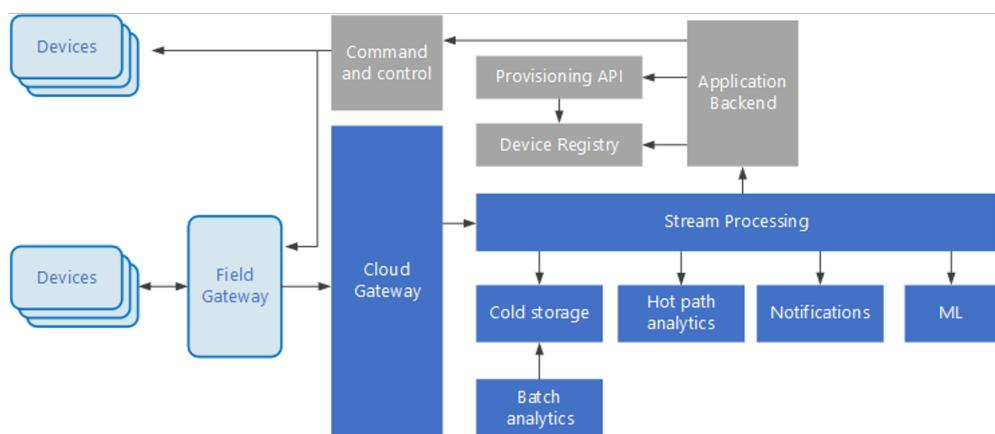
- ASVSP - Primeri arhitektura - IoT
 - arhitektura zasnovana na obradi događaja (engl. *event-driven architecture*)
 - npr. priključivanje, isključivanje i generisanje podataka od strane velikog broja IoT uređaja
 - lokalni agregatori podataka (engl. *field gateway*)
 - skupljaju podatke od grupa uređaja koji su sa njima povezani
 - mogu filtrirati ili transformisati podatke i protokole
 - šalju podatke globalnom agregatoru podataka
 - globalni aggregator podataka (engl. *cloud gateway*)
 - implementiran kao sistem za razmenu poruka sa brzim odzivom i malim kašnjenjima
 - prihvata podatke od svih uređaja
 - prosledjuje podatke obradivačima tokova podataka
 - koji obrađuju podatke i smeštaju ih po potrebi u druge komponente ASVSP-a

87

87

Arhitekture sistema velikih skupova podataka

- ASVSP - Primeri arhitektura - IoT



izvor: *Big data architectures*, Microsoft, <https://docs.microsoft.com/en-us/azure/architecture/data-guide/big-data/>

88

88

Principi projektovanja ASVSP

89

Principi projektovanja ASVSP

1. Projektovati osnovne komponente ASVSP pažljivo
2. Projektovati ASVSP za rad u prisustvu otkaza
3. Projektovati sisteme koji skaliraju
4. Projektanti su vođe projekata
5. Projektovati ASVSP bez prestanka
6. Projektovati slabo spregnute sisteme
7. Donositi odluke sa poništivim efektima
8. Projektovati i razvijati ASVSP uz prioritetizaciju bezbednosti
9. Projektovati ASVSP imajući cenu izvršavanja na umu

90

1. Projektovati osnovne komponente ASVSP pažljivo

- Jedan od osnovnih zadataka inženjera podataka da odabere osnovne komponente koje će biti korišćene u organizaciji
 - osnovne i zajedničke komponente koje se koriste na svim projektima
 - dozvoljavaju prevazilaženje **silosa** i omogućavaju **deljenje znanja i veština**
- Infrastrukturna rešenja koja uvećavaju agilnost timova
 - posebna pažnja mora biti usmerena ka podršci bezbednosnih mehanizama
 - koja zadovoljavaju zahteve svih projekata
 - često, ali ne nužno, biraju se rešenja u oblaku
- Potrebno postići balans između
 - **generičnosti rešenja** – koja su dovoljno fleksibilna da podrže višestruke projekte i nadogradnje
 - **posebnosti rešenja** – koja omogućavaju optimalno rešenje u nekom uskom domenu primene

91

91

2. Projektovati ASVSP za rad u prisustvu otkaza

- „**Sve će kad-tad otkazati!**“ (engl. “*Everything fails*”, Werner Vogels, CTO, AWS)
- Projektovanje ASVSP mora uključiti reakciju na otkaze
 - iako je suvremenih hardvera poprilično robustan
 - kad-tad će i u najboljem hardveru doći do otkaza
 - posebno jer su ovakvi sistemi po svojoj prirodi distribuirani
 - izvršavaju se na velikom broju hardverskih uređaja
 - povećava se verovatnoća otkaza sa porastom broja hardverskih uređaja
- Potrebno pratiti sledeće karakteristike i spram željenih osobina donositi arhitekturne odluke
 - **dostupnost** (engl. *Availability*)
 - **pouzdanost** (engl. *Reliability*)
 - **ciljna tačka oporavka** (engl. *recovery point objective*, RPO)
 - **ciljno vreme oporavka** (engl. *recovery time objective*, RTO)

92

92

3. Projektovati sisteme koji skaliraju

- **Skaliranje** (engl. *scalability*) sistema obuhvata dve aktivnosti
 - **dodavanje resursa** – kako bi sistem mogao da odgovori na rastući broj zahteva i količinu podataka
 - **uklanjanje resursa** – kako bi sistem oslobodio nekorišćene resurse usled smanjenja obima posla
 - neki sistemi mogu da uklone i sve resurse iz upotrebe, tj. *skaliraju do nule*
- **Elastični sistem** (engl. *elastic system*) – sistem koji je u mogućnosti dinamički da dodaje i uklanja resurse, tj. dinamički da skalira
 - u idealnom slučaju na automatizovan način
- Odabir strategije skaliranja u procesu projektovanja sistema je ključan
 - neadekvatna strategija može da rezultuje
 - nepotrebno komplikovanom arhitekturom
 - visokom cenom izvršavanja
 - vrlo često zahteva precizna merenja opterećenja sistema, količine podatka, broja zahteva i slično

93

93

4. Projektanti su vođe projekata

- Projektanti ASVSP su zaduženi za
 - odluke po pitanju **odabira tehnologija u uspostavljanju arhitekture**
 - komunikaciju odluka sa ostalim članovima tima kroz diskusiju, efektivno **vođstvo i obuku**
 - obuku ostalih članova po pitanju tehnoloških odluka
 - koje za cilj imaju unapređenje tehnološke zrelosti kompanije i unapređenje njenog poslovanja
- Projektanti ASVSP su **tehnički veoma potkovani**
 - često delegiraju implementaciju ostalim članovima tima
 - poseduju **veštine vođenja** timova i projekata
 - veoma poželjna **dualnost**

94

94

5. Projektovati ASVSP bez prestanka

- Sistem se razvija kako se razvija i proizvod ili organizacija u okviru koje je implementiran
 - projektanti moraju da prate metrike izvršavanja sistema
 - projektanti moraju da prate razvoj tehnologija
 - projektanti moraju da prate metrike poslovanja organizacije
 - projektanti prilagođavaju ASVSP promenama uočenim u prethodno navedenim elementima
- Projektovanje je kontinualni proces koji nema kraj
- Projektanti su dužni da
 - detaljno poznaju **trenutnu ASVSP**
 - kontinualno razvijaju **ciljnu arhitekturu ASVSP**
 - razviju **detaljan plan** kako od trenutnu ASVSP promeniti da se dobije ciljna ASVSP
 - sa što manje uticaja na trenutno poslovanje

95

95

6. Projektovati slabo spregnute sisteme

- **Slabo sprezanje** ima i **tehnološke i organizacione** konotacije
 - sistemi moraju imati slabo spregnute tehnološke komponente
 - što doprinosi lakšem razvoju, testiranju i skaliranju
 - razvoj se dodeljuje **manjim timovima**
 - svaki tim odgovoran je za neku komponentu sistema
 - timovi za svoje komponente javno objavljuju isključivo API ili šemu podataka
 - bez bilo kakvih detalja implementacije
 - omogućava **slabo spregnutu komunikaciju** između timova
 - timovi mogu brzo da razvijaju i unapređuju svoje komponente
 - nezavisno od ostalih timova (posledica slabo spregnute komunikacije)
 - timovi mogu da objavljuju svoje komponente sa **minimalnom nedostupnosti sistema**
 - neposredno omogućava **donošenje poništivih odluka**

96

96

6. Projektovati slabo spregnute sisteme

- Primer koji je omogućio razvoj AWS-a (*Bezos API Mandate 2002*)
 - <https://nordicapis.com/the-bezos-api-mandate-amazons-manifesto-for-externalization/>
 - 1. **All teams will henceforth expose their data and functionality through service interfaces.**
 - 2. **Teams must communicate with each other through these interfaces.**
 - 3. **There will be no other form of interprocess communication allowed:** no direct linking, no direct reads of another team's data store, no shared-memory model, no back-doors whatsoever. The only communication allowed is via service interface calls over the network.
 - 4. **It doesn't matter what technology they use.** HTTP, Corba, Pubsub, custom protocols — doesn't matter.
 - 5. **All service interfaces, without exception, must be designed from the ground up to be externalizable.** That is to say, the team must plan and design to be able to expose the interface to developers in the outside world. No exceptions.
 - 6. Anyone who doesn't do this will be fired.
 - 7. *Thank you; have a nice day!*

97

97

7. Donositi odluke sa poništivim efektima

- Dve vrste odluka
 - Jeff Bezos <https://www.inc.com/jeff-haden/amazon-founder-jeff-bezos-this-is-how-successful-people-make-such-smart-decisions.html>
 - odluke tipa 1 – vrata sa kvakom sa jedne strane
 - jednom kada se ovakve odluke donesu nema povratka na prethodno stanje
 - odluke tipa 2 – vrata sa kvakama na obe strane
 - odluke koje se mogu poništiti, tj. gde je moguće vratiti se na prethodno stanje
 - uvek treba težiti donošenju odluka tipa 2
 - većina odluka jeste tipa 2
 - postoji problem pogrešnog određivanja tipa odluke
- Donošenje poništivih odluka
 - omogućava fleksibilnost u razvoju softverskih sistema
 - usled konstantnog razvoja tehnologija
 - usled konstantne pojave novih pristupa razvoju softvera
 - omogućava brzu evoluciju sistema

98

98

7. Donositi odluke sa poništivim efektima

- „Jedan od osnovnih projektantskih zadataka jeste eliminisanje potrebe za detaljnim specificiranjem arhitekture softverskog sistema pronalaženjem načina da se prevaziđe neponištivost odluka u samom procesu projektovanja softvera“
 - Martin Fowler, “Who needs an Architect”
 - <https://martinfowler.com/ieeeSoftware/whoNeedsArchitect.pdf>

99

99

8. Projektovati i razvijati ASVSP uz prioritetizaciju bezbednosti

- Svaki projektant i inženjer podataka **je odgovoran** za bezbednost sistema koji razvija
 - projektovati komponente sistema da budu bezbedne
 - koristiti elemente platforme prema principima bezbednog korišćenja
- **Model bezbednosti nultog poverenja** (engl. zero-trust security model)
 - zahtevati eksplicitnu dozvolu za pristup servisu ili podacima
 - neovisno od "pozicije" onoga ko pristup zahteva (interno ili eksterno)
 - rešava problem koji postoji kod tradicionalnih pristupa
 - kao što je pristup obezbeđenju granica sistema (engl. hardened-perimeter model)
 - koji onemogućavaju pristup samo spoljnjim entitetima
 - podloži napadima iznutra
- **Model deljene odgovornosti** (engl. shared-responsibility model)
 - pružalac platformskih usluga (u oblaku) je odgovoran za bezbednost platforme
 - korisnik platformskih usluga odgovoran je za bezbednost aplikacije koja se izvršava

100

100

9. Projektovati ASVSP imajući cenu izvršavanja na umu

- Tradicionalno projektovanje ASVSP
 - obuhvata projektovanje hardversko-softverske infrastrukture da podrži što bolje performanse
- Savremeno projektovanje ASVSP
 - obuhvata **balansiranje između performansi i cene platforme** na kojoj se sistem izvršava
 - podržan modelom pružaoca usluga u oblaku: „plati koliko koristiš”
 - zahteva detaljno poznavanje platforme, tehnologije ali i finansijskih mogućnosti organizacije
- Nastaje **FinOps** kao disciplina
 - *FinOps Foundation* <https://www.finops.org/introduction/what-is-finops/>

101

101

Principi projektovanja arhitektura sistema u oblaku

- Pošto se vežina ASVSP izvršava u oblaku, potrebno je ispoštovati i principe projektovanja arhitektura sistema koji se namenski prave za izvršavanje u oblaku
- *AWS Well-Architected Framework*
 - *Operational excellence*
 - *Security*
 - *Reliability*
 - *Performance efficiency*
 - *Cost optimization*
 - *Sustainability*
 - <https://aws.amazon.com/architecture/well-architected/>

102

102

Principi projektovanja arhitektura sistema u oblaku

- Pošto se vežina ASVSP izvršava u oblaku, potrebno je ispoštovati i principe projektovanja arhitektura sistema koji se namenski prave za izvršavanje u oblaku
- *Google's 5 principles for cloud-native architecture*
 - *Design for automation*
 - *Be smart with state*
 - *Favour managed services*
 - *Practice defence in depth*
 - *Always be architecting*
 - <https://cloud.google.com/blog/products/application-development/5-principles-for-cloud-native-architecture-what-it-is-and-how-to-master-it>

103

103

Primena i izazovi ASVSP

104

Primena i izazovi ASVSP

- ASVSP
 - razmotriti primenu ASVSP u slučajevima kada je potrebno
 - skladištitи i obraditi veliku količinu podataka
 - preveliku za tradicionalne baze podataka
 - transformisati nestrukturirane podatke za dalju analizu i izveštavanje
 - skladištitи, obraditi i analizirati neprekidan tok podataka u realnom vremenu
 - ili sa veoma malim vremenom trajanja obrade

105

105

Primena i izazovi ASVSP

- Slučajevi korišćenja ASVSP
 - **razvoj proizvoda u kompanijama**
 - kreiraju se prediktivni modeli za nove proizvode i servise koje kompanije nude
 - klasifikacija po ključnim obeležjima starih i novih proizvoda
 - modelovanje veze između tih atributa i komercijalnog uspeha proizvoda
 - **prediktivno održavanje uređaja**
 - predikcija održavanja uređaja u fabrikama
 - pre nego što se oni pokvare
 - omogućava naručivanje delova unapred kao i preventivne popravke
 - posmatranjem karakteristika uređaja, logova, podataka sa senzora i opisa i karakteristika prethodnih kvarova svih uređaja

106

106

Primena i izazovi ASVSP

- Slučajevi korišćenja ASVSP
 - **unapređenje korisničkog iskustva**
 - skupljanjem velike količine podataka o korisniku moguće je
 - ponuditi mu prilagođene usluge i proizvode
 - sprečiti pojavu problema proaktivnim delovanjem
 - **unapređenje sigurnosti**
 - uočavanje šablonu koji su indikatori prevare
 - skupljanje velike količine podataka kako bi se omogućilo slanje što preciznijih izveštaja
 - obično vlastima, gde je to zahtevano
 - **mašinsko učenje**
 - kvalitetno obučavanje algoritama za mašinsko učenje zahteva velike količine podataka
 - što veći broj uzoraka nad kojima algoritam uči

107

107

Primena i izazovi ASVSP

- Slučajevi korišćenja ASVSP
 - **efikasnost operative**
 - analiza proizvodnje, povratnih informacija od korisnika i problema i vraćanja proizvoda
 - mogu rešiti potencijalne probleme u operativi pre nego što nastanu
 - mogu predvideti buduće potrebe tržišta
 - mogu uticati na poboljšanje odluka menadžmenta u skladu sa predviđenim i trenutnim kretanjima tržišta
 - **pokretanje inovacija u kompanijama**
 - analizom zavisnosti između ljudi, institucija, procesa i drugih entiteta
 - koji se mogu uočiti u prikupljenim podacima
 - može dovesti do
 - novih odluka o ulaganjima
 - boljih finansijskih planova

108

108

Primena i izazovi ASVSP

- Slučajevi korišćenja ASVSP
 - **unapređenje zdravstvene zaštite**
 - unapređenje procesa otkrivanja bolesti
 - identifikacija ranih znakova bolesti i elemenata koji utiču na njenu pojavu i napredovanje
 - razvoj novih lekova
 - **predviđanje i reagovanje na prirodne i nepogode izazvane od strane čoveka**
 - analiza podataka sa postavljenih senzora
 - za predviđanje zemljotresa i vulkanskih aktivnosti
 - analiza šablona ljudskog ponašanja
 - za pomoći prilikom organizovanja pomoći preživelima ili ugroženima
 - za sigurno preseljavanje izbeglica izvan zona ratnih dejstava
 - **sprečavanje zločina**
 - na osnovu analize dosjeda, prethodnih slučajeva, snimaka sigurnosnih kamera
 - moguće bolje rasporediti policijske snage
 - moguće bolje preventivno reagovati gde je to potrebno

109

109

Primena i izazovi ASVSP

- Izazovi ASVSP
 - **privatnost podataka**
 - čuvanje osetljivih i ličnih podataka o nekom pojedincu ili kompanije
 - Koliko dugo je dozvoljeno takve podataka čuvati?
 - koliko je potrebno takvih podataka čuvati?
 - GDPR <https://euqdpr.org/>
 - **sigurnost podataka**
 - mehanizmi za sprečavanje neovlašćenog pristupa osetljivim i ličnim podacima
 - **diskriminacija uzrokovana podacima**
 - moguće je namerno ili nenamerno izvršiti diskriminaciju nad tom osobom
 - nakon sakupljanja velike količine ličnih podataka o osobi
 - primeri: osiguranje i kreditiranje

110

110

Primena i izazovi ASVSP

- Izazovi ASVSP
 - **količina podataka koja se konstantno uvećava**
 - količina podataka se udvostručuje svake dve godine
 - potrebno efikasno i dovoljno veliko skladište podataka
 - **verodostojnost podataka**
 - posedovanje čistih i relevantnih podataka
 - staranje o podacima postaje imperativ a zahteva dosta vremena i resursa
 - osobe koje rade sa podacima, provedu 50-80 posto vremena samo čisteći podatke i pripremajući ih za analizu
 - **praćenje razvoja tehnologija**
 - veliki broj projekata i pristupa se razvija
 - bez garancije da će i jedan od njih biti uspešan
 - ali sa mogućnošću da će neki zameniti trenutno *de facto* standardne alate i pristupe

111

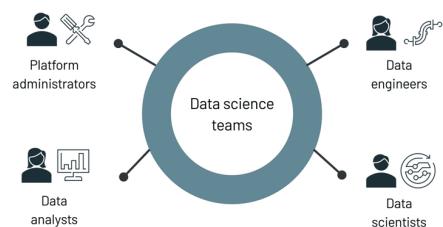
111

Uloge u timu za rad sa podacima

112

Uloge u timu za rad sa podacima

- Uloge u timu za rad sa podacima
 - više članova tima za rad sa podacima
 - usled kompleksnosti sistema i složenosti zadataka
 - članovi tima mogu biti tehnička ali i netehnička lica
 - u malim organizacijama, ne mora postojati različita osoba za svaku ulogu
 - četiri različite uloge u timovima koji rade sa podacima:
 - **administratori platforme** (engl. *platform administrators*)
 - **inženjeri podataka** (engl. *data engineers*)
 - **analitičari podataka** (engl. *data analysts*)
 - **istraživači podataka** (engl. *data scientists*)



113

izvor: Fundamentals of Big Data, Databricks, <https://academy.databricks.com/>

113

Uloge u timu za rad sa podacima

- Uloge u timu za rad sa podacima
 - **administratori platforme**
 - održavaju infrastrukturu nad kojom se izvršava ASVSP
 - nadgledanje, konfiguracija, ažuriranje
 - IT podrška ostalim članovima tima a u vezi s ASVSP
 - **inženjeri podataka**
 - kreiraju i održavaju strukture i mehanizme za rad sa podacima
 - omogućavaju da podaci dolaze do potrebnih osoba ili organizacija
 - definišu „aplikativni“ sloj koji koriste analitičari i istraživači podataka

114

114

Uloge u timu za rad sa podacima

- Uloge u timu za rad sa podacima
 - **analitičari podataka**
 - izvlače informacije iz podataka pripremljenih od strane inženjera podataka
 - pripremaju prezentaciju podataka koja se sastoji od grafikona itd.
 - obučeni da kreiraju vizuelizacije podataka i poznaju koncepte poslovnog izveštavanja
 - komuniciraju sa rukovodstvom u cilju izvlačenja uvida i semantike podataka i njihovog značaja za poslovno okruženje
 - **istraživači podataka**
 - izvlače informacije iz podataka pripremljenih od strane inženjera podataka
 - koriste matematičke i statističke metode, kreiraju modele mašinskog učenja
 - obučeni kako matematičkim i statističkim veštinama tako i programerskim

115

115

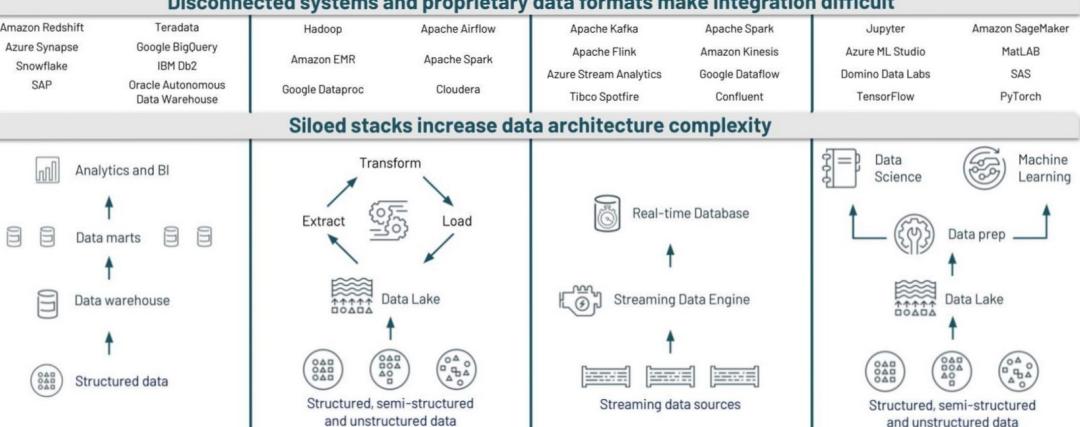
Uloge u timu za rad sa podacima



Siloed data teams decrease productivity

Disconnected systems and proprietary data formats make integration difficult

Siloed stacks increase data architecture complexity



116

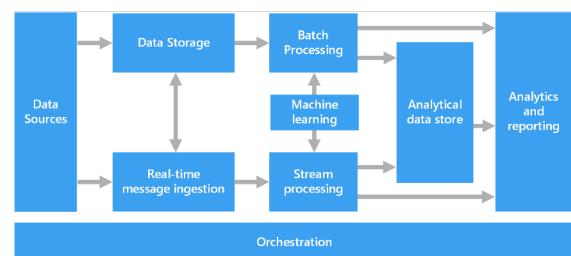
116

Sadržaj kursa

117

Arhitekture sistema velikih skupova podataka

- Sadržaj kursa - glavne teme
 - **virtualizacija, kontejnerizacija i orkestracija platforme**
 - Docker, Docker-Compose, Kubernetes
 - **paketna obrada podataka**
 - Hadoop, MapReduce i Spark
 - **obrada podataka u realnom vremenu**
 - međuprocesna komunikacija i sistemi za razmenu poruka
 - Kafka
 - Storm i Spark
 - **arhitekture Serverless**
 - **arhitekture Lambda i Kappa**



izvor: *Big data architectures*, Microsoft, <https://docs.microsoft.com/en-us/azure/architecture/data-guide/big-data/>¹¹⁸

118

Primeri projekata

119

Projekat - Dinamičko pronalaženje putanje

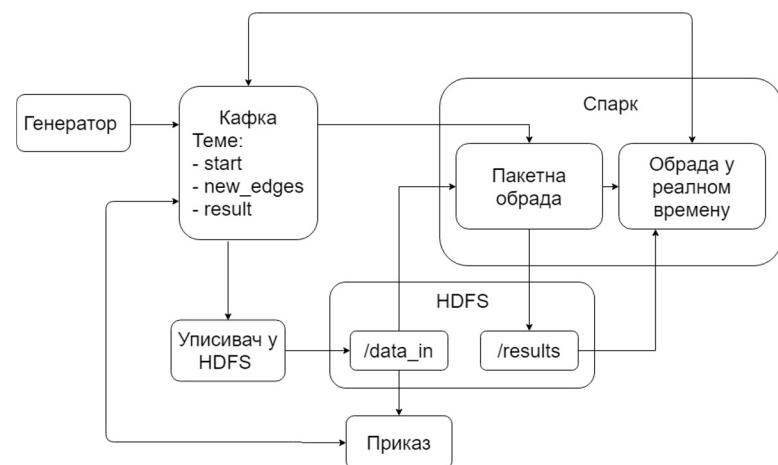
- Dejan Grubišić, ASVSP, MSc rad
 - sistem za pronalaženje najkraće putanje između lokacija
 - sa težinama putanja promenljivim u realnom vremenu
 - zasnovan na arhitekturi Lambda
 - primenljiv na veliki broj domena u kojima su podaci predstavljeni pomoću grafa
 - npr. pronalaženje najboljeg puta između traženih lokacija i njegovo ažuriranje u realnom vremenu
 - paketna obrada prethodno skladištenih informacija
 - npr. informacija o putevima na mapi koja se posmatra
 - radi inicijalnog pronalaženja najboljeg puta
 - obrada u realnom vremenu informacija koje pristižu
 - npr. saobraćajne nezgode, blokirani semafori, kolone itd.
 - radi ažuriranja rang liste puteva

120

120

Projekat - Dinamičko pronalaženje putanje

- Arhitektura sistema

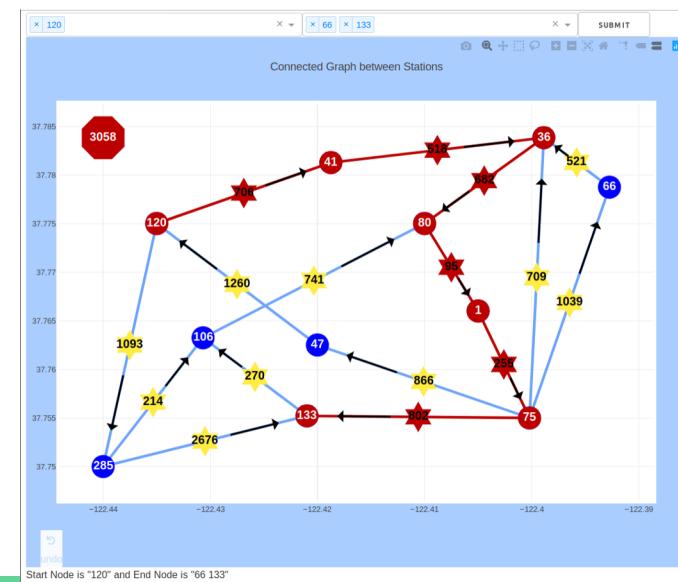


121

121

Projekat - Dinamičko pronalaženje putanje

- Interfejs sistema

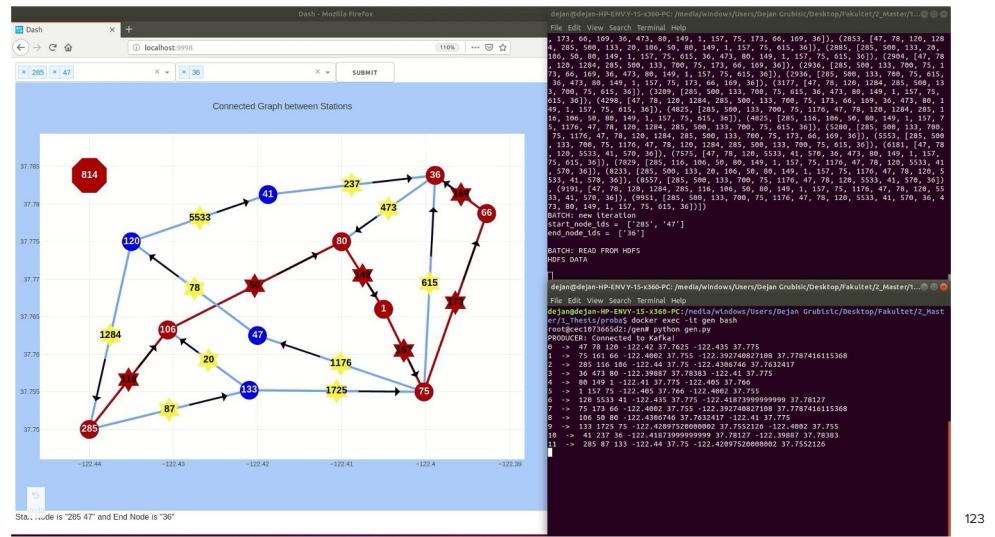


122

122

Projekat - Dinamičko pronalaženje putanje

- Demo



123

Projekat – Sistem za analizu podataka Formule 1

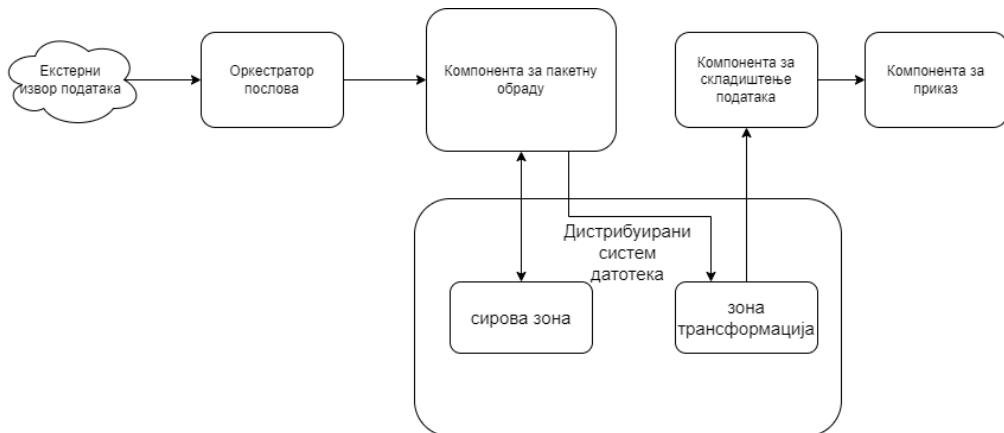
- Aleksa Vučaj, ASVSP, MSc rad
 - cilj rada - omogućiti lakšu analizu i vizuelizaciju podataka telemetrije Formule 1
 - sistem za obradu i analizu telemetrijskih podataka Formule 1
 - paketna obrada istorijskih podataka
 - distribuirani sistem datoteka Hadoop (engl. *Hadoop Distributed File System*, HDFS)
 - obrada podataka u alatu Apači Spark (engl. *Apache Spark*)
 - trajno skladištenje obrađenih podataka u bazi podataka Postgre (engl. *PostgreSQL*)
 - orkestracija tokova poslova nad kojim je implementirana komponenta za periodično dobavljanje novih podataka pomoću Apači Erflou (engl. *Apache Airflow*)
 - BI alat za interaktivnu analizu podataka
 - za prikaz rezultata obrade iskorišćen je alat za vizuelizaciju podataka Metabejz (engl. *Metabase*)
 - svi pomenuti delovi sistema upakovani su u zasebne kontejnere kreirane korišćenjem tehnologije Docker (engl. *Docker*)

124

124

Projekat – Sistem za analizu podataka Formule 1

- Aleksa Vučaj, ASVSP, MSc rad



125

125

Projekat – Sistem za analizu podataka Formule 1

- Aleksa Vučaj, ASVSP, MSc rad



126

126

Projekat – Sistem za analizu podataka Formule 1

- Aleksa Vučaj, ASVSP, MSc rad



127

127

Projekat - Sistem za analizu nepogoda u USA

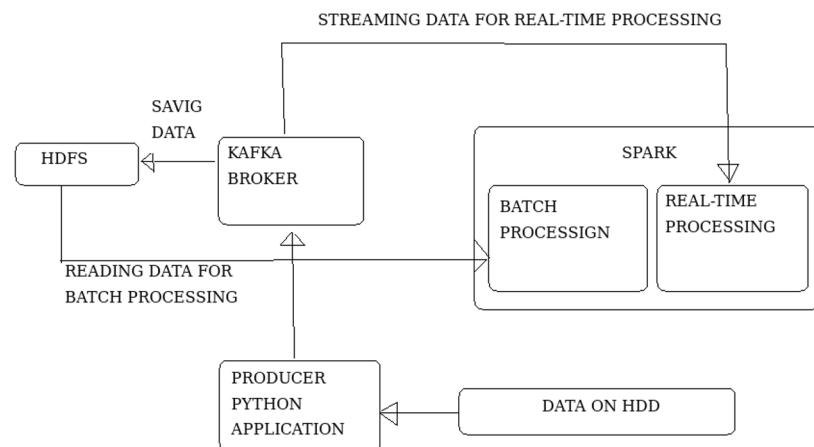
- Nebojša Horvat, ASVSP, projekat
 - sistem za analizu nepogoda u USA
 - javno dostupan skup podataka
 - nepogode obuhvataju jače kiše, uragane, poplave sa materijalnom štetom i smrtnim ishodima
 - paketna obrada prethodno skladištenih informacija
 - dugotrajne analize
 - npr. broj žrtava i šteta po tipu nepogode, najgore nepogode koje su se dogodile, nepogode koje predstavljaju outlier-e
 - obrada u realnom vremenu informacija koje pristižu
 - npr. brojač i alarm za obaveštavanje o nepogodama koje se dešavaju

128

128

Projekat - Sistem za praćenje nepogoda u USA

- Arhitektura sistema



129

129

Literatura

- Joe Reis, Matt Housley – *Fundamentals of Data Engineering: Plan and Build Robust Data Systems*
- Martin Kleppmann - *Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems*
- Nathan Marz, James Warren - *Big Data: Principles and best practices of scalable real-time data systems*
- Doug Laney - *3D Data Management: Controlling Data Volume, Veracity, and Variety*
<https://blogs.gartner.com/doug-laney/files/2012/01/ad949-3D-Data-Management-Controlling-Data-Volume-Velocity-and-Variety.pdf>

130

130