

Велики скупови података

Лекција 7

Стеван Гостојић

Факултет техничких наука, Нови Сад

12. мај 2022.

Преглед садржаја

- 1 Увод
- 2 MapReduce & Hadoop
- 3 Publish-subscribe & Kafka
- 4 Закључак

Преглед садржаја

- 1 Увод
- 2 MapReduce & Hadoop
- 3 Publish-subscribe & Kafka
- 4 Закључак

Велики скупови података

- Велики скупови података (енг. big data) је појам који се односи на скупове података који се не могу складиштити и обрадити "традиционалним" методама и алатима
- Упознаћемо се са основним појмовима и принципима (складиштења и обраде) великих скупова података, а детаљи имплементације метода и алата за складиштење и обраду великих скупова података биће обрађени на другим предметима
- Складиштење и обрада великих скупова података биће илустровани на платформама Hadoop и Kafka

Велики скупови података

- Повећање капацитета за складиштење, пренос и обраду података је проузроковала повећање података у (бар) три димензије: запремини (енг. volume), брзини (енг. velocity) и разноликости (енг. variety)
- "3V"

Big Data: 3V's

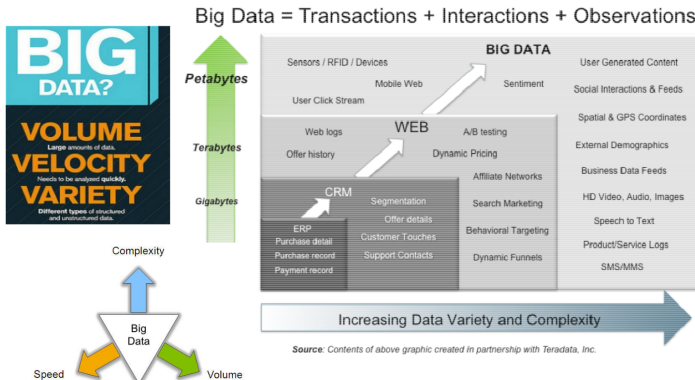


Figure 1: 3V.

Volume

- Величина скупа података: \sim ТВ

Velocity

- Брзина стварања (или промене) података: $\sim \text{Gb/s}$

Variety

- Тип скупа података: текст, слика, аудио запис, видео запис итд.

Variety

- структурирани подаци (нпр. релационе базе података, JSON документи, RDF графови итд.)
- полуструктурирани подаци (нпр. XML документи)
- неструктурирани подаци (нпр. текстуални документи, фотографије, аудио записи, видео записи итд.)

Примена

- Друштвени медији
- Сензори
- Интернет претраге
- Берзанске трансакције
- Астрономија
- Медицина
- итд.

Хронологија

- Online Transaction Processing (OLTP) - DBMSs
- Online Analytical Processing (OLAP) - Data Warehousing
- Real-Time Analytics Processing (RTAP) - Big Data Architecture & Technology

Велики скупови података

- Методе и алати за обраду великих скупова података имају одређене мане (тј. не обезбеђују иста својства која обезбеђују методе и алати за обраду малих скупова података)
- BASE уместо ACID

ACID

- Атомичности (енг. atomicity) - ако било која операција не успе, цела трансакција не успе
- Конзистентност (енг. consistency) - база података увек прелази из једног конзистентног стања у друго конзистентно стање
- Изолација (енг. isolation) - више паралелних трансакција не утичу једна на другу
- Трајност (енг. durability) - након што се излаз операције запише у базу података, остаје у бази података чак и после губитка напајања, пада система или других грешака

BASE

- У основи доступно (енг. basically available) - обезбеђује доступност података реплицирањем по чворовима кластера
- Меко стање (енг. soft state) - због недостатка конзистентности, подаци могу да се промене током времена (а одговорност за обезбеђивање конзистентности података се делегира на програмера)
- На крају конзистентно (енг. eventual consistency) - чињеница да BASE не обезбеђује непосредну конзистентност не значи да је никада не постиже (читање података је могуће иако можда не одражава стварност)

Складиштење великих скупова података

Постоје различити модели складиштења великих скупова података:

- Дистрибуирани системи датотека
- итд.

Обрада великих скупова података

Постоје различити модели обраде великих скупова података:

- Обрада серија података (енг. batch processing)
- Обрада токова података (енг. stream processing)
- итд.

Обрада серија података

- Код обраде серија података (енг. batch processing), подаци се прво складиште, а онда обрађују
- Намењена је обради великих скупова података која није временски захтевна

Обрада серија података

- Обрада серија података може се посматрати и као модел покретања програма
- Интеракција са корисником се своди на предају послова који треба да се изврше од стране корисника систему
- Послови могу да се аутоматски извршавају у заказано време или у зависности од доступности рачунарских ресурса

Обрада серија података

- Постоје различити програмски модели за обраду серија података
- Вероватно најпознатији програмски модел је MapReduce

Обрада токова података

- Код обраде токова података (енг. stream processing) подаци се обрађују у реалном времену (док улазе у систем)
- Намењена је обради великих скупова података чији резултати су потребни одмах

Обрада токова података

- Постоје различити програмски модели за обраду токова података
- Вероватно најпознатији програмски модел је објави-претплати се (енг. publish-subscribe)

Преглед садржаја

- 1 Увод
- 2 MapReduce & Hadoop
- 3 Publish-subscribe & Kafka
- 4 Закључак

MapReduce

- MapReduce је програмски модел за паралелну обраду велике количине структурираних података
- Неке од особина овог програмског модела су дистрибуираност (над великим бројем јефтиних процесора), скалабилност (број процесора се по потреби може повећавати или смањивати) и отпорност на грешке (обрада се може наставити упркос неким грешкама)
- Настао је са циљем да се програми написани по овом моделу извршавају на кластерима јефтиних рачунара

MapReduce

- MapReduce програмски модел укључује две фазе: фазу мапирања (енг. map) и фазу свођења (енг. reduce)

Фаза мапирања

- Фаза мапирања (енг. map) трансформише улазни скуп података у излазни скуп података (у облику парова кључ/вредност)

Фаза свођења

- Фаза свођења (енг. reduce) трансформише улазни скуп података (који је излаз из фазе мапирања) у излазни скуп података (који је мањи скуп парова кључ/вредност)

MapReduce

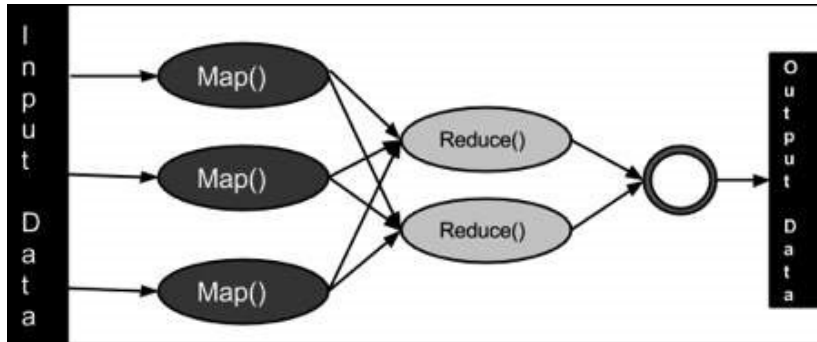


Figure 2: MapReduce.

Пример

- Потребно је израчунати максималну температуру у Београду, Новом Саду и Нишу на основу података о измереним температурама
- Измерене температуре у различитим градовима у различитим временским тренутцима су складиштене у текстуалним датотекама (за сваки датум појединачно)

Пример

1	2022-05-11T14:00	Palic	25	1007.1	SSW	3	39	25
2	2022-05-11T14:00	Sombor	27	1009.1	W	3	21	27
3	2022-05-11T14:00	Novi Sad	24	1010.0	VNW	2	47	24
4	2022-05-11T14:00	Zrenjanin	25	1010.3	W	3	41	25
5	2022-05-11T14:00	Kikinda	26	1010.2	SSW	2	28	26
6	2022-05-11T14:00	B. Karlovac	25	1009.6	VNW	2	54	25
7	2022-05-11T14:00	Loznica	24	1005.5	WSW	2	34	24
8	2022-05-11T14:00	S. Mitrovica	25	1010.6	S	2	39	25
9	2022-05-11T14:00	Valjevo	24	999.8	N	2	41	24
10	2022-05-11T14:00	Beograd	26	1004.6	W	2	34	26
11	2022-05-11T14:00	Kragujevac	24	999.1	N	2	44	24
12	2022-05-11T14:00	S. Palanka	24	1006.1	N	1	41	24
13	2022-05-11T14:00	V. Gradiste	25	1010.1	SW	1	32	25
14	2022-05-11T14:00	Crni Vrh	18	902.8	NW	3	45	18
15	2022-05-11T14:00	Negotin	25	1013.9	WSW	2	41	25
16	2022-05-11T14:00	Zlatibor	20	904.6	ENE	2	43	20
17	2022-05-11T14:00	Sjenica	19	903.3	NW	2	40	19
18	2022-05-11T14:00	Pozega	24	983.9	E	1	36	24
19	2022-05-11T14:00	Kraljevo	24	994.7	N	2	38	24
20	2022-05-11T14:00	Kopaonik	10	833.6	E	2	71	10
21	2022-05-11T14:00	Kursumlija	24	975.4	ENE	3	27	24
22	2022-05-11T14:00	Krusevac	25	1000.2	NW	2	39	25
23	2022-05-11T14:00	Cuprija	26	1005.4	NW	2	39	26
24	2022-05-11T14:00	Nis	25	995.9	NW	2	34	25
25	2022-05-11T14:00	Leskovac	25	992.6	SW	1	32	25
26	2022-05-11T14:00	Zajecar	26	1002.4	VNW	2	28	26
27	2022-05-11T14:00	Dimitrovgrad	20	967.5	E	5	56	20
28	2022-05-11T14:00	Vranje	24	969.2	NE	2	24	24

29

Фаза мапирања

- Ангажујмо 100 тар процесора да на основу улазног скупа датотека израчунају излазни скуп уређених парова
- Сваком тар процесору ћемо доделити одређени број улазних датотека, па они могу да раде у паралели

Пример

```
1 ( Novi Sad , 24)
2 ( Beograd , 26)
3 ( Nis , 25)
4 ( Novi Sad , 23)
5 ( Beograd , 25)
6 ( Nis , 24)
7 ( Novi Sad , 25)
8 ( Beograd , 26)
9 ( Nis , 26)
10 .
11 .
12 .
13
```


Фаза свођења

- Ангажујмо 3 reduce процесора (по један за Београд, Нови Сад и Ниш) да на основу улазног скупа парова израчунају излазни скуп максималних вредности
- Сваком reduce процесору ћемо доделити уређене парове са одређеним кључем, па они могу да раде у паралели

Пример

- 1 (Novi Sad, 25)
- 2 (Beograd, 26)
- 3 (Nis, 26)
- 4

Hadoop

- Hadoop је имплементација MapReduce програмског модела отвореног изворног кода
- Написана је у Јави
- Основне компоненте Hadoop-а су Hadoop Distributed File System (HDFS) и MapReduce Engine, а помоћне компоненте Hadoop YARN и Hadoop Common

HDFS

- Hadoop Distributed File System (HDFS) је компонента за дистрибуирано складиштење (велике количине) података
- HDFS захтева да се подаци поделе у блокове (подразумевана величина блока је 64 MB)
- Због редувантности се сваки блок складишти на два или више чворова (у различитим рековима)

MapReduce Engine

- MapReduce Engine је компонента која извршава програме написана по MapReduce програмском моделу

Hadoop YARN

- Hadoop YARN је радни оквир за заказивање послова и управљање ресурсима кластера

Hadoop Common

- Hadoop Common је скуп библиотека и помоћних програма које захтевају друге Hadoop компоненте

Hadoop кластер

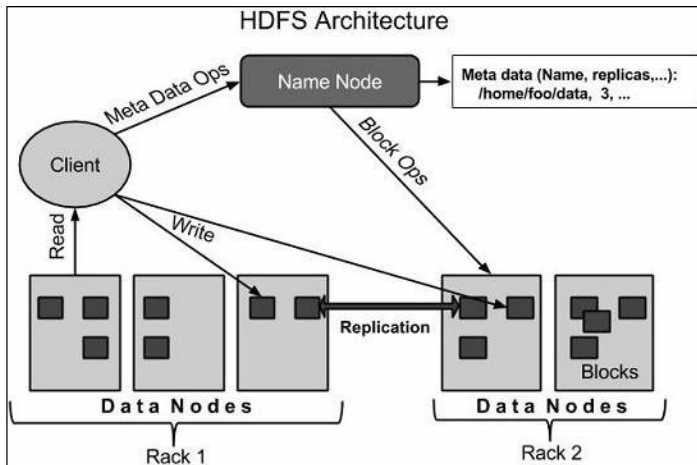


Figure 3: Hadoop.

Нadoop кластер

- Чвор имена (енг. name node) - управља простором имена система датотека (тј. прати где се налази сваки блок)
- Чвор података (енг. data node) - складишти податке у систему датотека
- Трагач послова (енг. job tracker) - додељује посао мапирања трагачу задатака који се налази близу података (у истом реку)
- Трагач задатака (енг. task tracker) - извршава посао што је могуће ближе подацима

Hadoop кластер

- Подаци се деле у директоријуме и датотеке, а датотеке се деле у блокове униформне величине (64 MB или 128 MB)
- Датотеке се дистрибуирају између различитих чворова кластера
- Блокови се реплицирају да би се омогућио опоравак од хардверских отказа
- HDFS, који се налази изнад локалног система датотека, надгледа дистрибуцију датотека

Нadoop кластер

- У фази мапирања се датотека обрађује на одређеном чвору
- Подаци се сортирају између фазе мапирања и фазе свођења
- У фази свођења се сортирани подаци шаљу на даљу обраду

Преглед садржаја

- 1 Увод
- 2 MapReduce & Hadoop
- 3 Publish-subscribe & Kafka**
- 4 Закључак

Објави-претплати се

- Модел објави-претплати се (енг. publish-subscribe) је програмски модел за обраду порука
- Издавачи (енг. publishers) шаљу поруке које припадају одређеној теми (енг. topic)
- Претплатници (енг. subscribers) примају поруке које припадају темама на које су се претплатили
- За дистрибуцију порука је задужен ред порука (енг. message queue)

Објави-претплати се

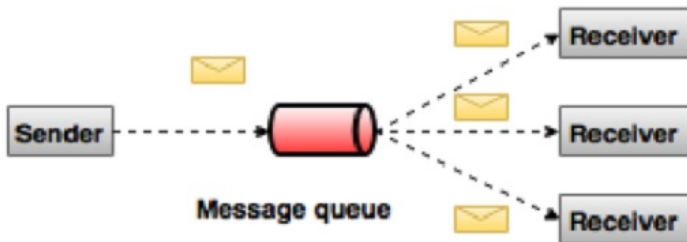


Figure 4: Објави-претплати се.

Kafka

- Kafka је дистрибуирани систем за обраду порука заснован на моделу објави-претплати се
- Дизајниран је да буде брз, скалабилан и отпоран на отказе
- Написан је у програмским језицима Scala and Java
- Основне компоненте Kafka-е су произвођачи (енг. producers), потрошачи (енг. consumers), посредници (енг. brokers), теме (енг. topics) и партиције (енг. partitions)

Kafka

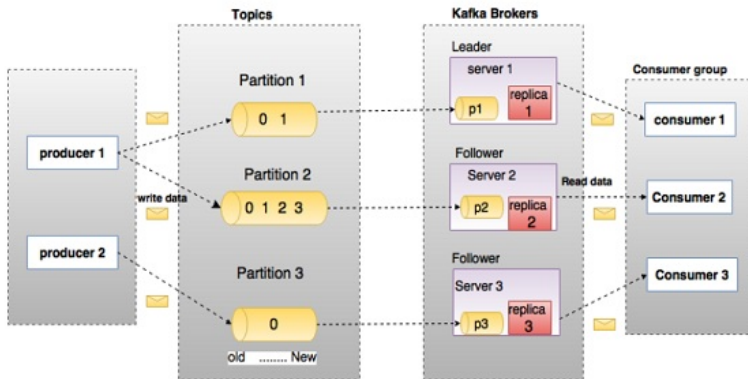


Figure 5: Kafka.

Kafka

- Теме (енг. topics) - токови порука које припадају одређеној категорији
- Партиције (енг. partitions) - теме се деле на партиције
- Посредници (енг. brokers) - системи одговорни за управљање објављеним подацима (сваки посредник може да садржи од нула до више партиција по теми)
- Кластери (енг. clusters) - кластери садрже више од једног посредника
- Произвођачи (енг. producers) - издавачи порука које припадају некој теми
- Потрошачи (енг. consumers) - читају поруке од посредника
- Вође (енг. leaders) - чворови кластера одговорани за сва читања и писање за дату партицују
- Пратиоци (енг. followers) - чворови који прати инструкције вође

Kafka

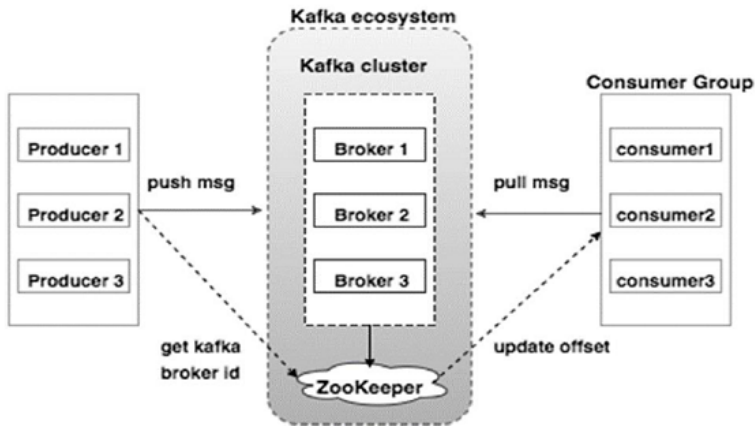


Figure 6: Kafka.

ZooKeeper

```
1 bin/kafka-server-start.sh config/server.properties
2
```

Topic

```
1 bin/kafka-topics.sh --create --zookeeper localhost:2181 --replication  
  --factor 1 --partitions 1 --topic Hello-Kafka
```

2

Producer

```
1 bin/kafka-console-producer.sh --broker-list localhost:9092 --topic  
  Hello-Kafka  
2
```

Producer

```
1 Hello
2 My first message
3 My second message
4
```

Consumer

```
1 bin/kafka-console-consumer.sh --zookeeper localhost:2181 --topic  
Hello-Kafka --from-beginning
```

```
2
```

Consumer

```
1 Hello
2 My first message
3 My second message
4
```


Kafka

- Претходни примери илуструју како се користе CLI алати за рад са Kafka-ом
- Са Kafka-ом се може да се ради (а обично се и ради) коришћењем API-ја (нпр. Python, Java, C# итд.)

Преглед садржаја

- 1 Увод
- 2 MapReduce & Hadoop
- 3 Publish-subscribe & Kafka
- 4 Закључак**

Закључак

- Велики скупови података
- ACID vs. BASE
- Обрада серија података
- Обрада токова података

Закључак

- Модел MapReduce
- Фаза мапирања
- Фаза свођења
- Hadoop
- HDFS
- MapReduce Engine
- Hadoop кластер

Закључак

- Модел објави-претплати се
- Издавач
- Претплатник
- Ред порука
- Kafka
- Произвођач
- Посредник
- Потрошач
- Kafka кластер

Литература

- Apache Hadoop, <https://hadoop.apache.org/>
- Apache Kafka, <https://kafka.apache.org/>

Хвала на пажњи!