

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 1954

**Stvarnovremensko praćenje
parametara ispravnosti rada u
sustavu za raspodijeljenu obradu
tokova podataka**

Mislav Jakšić

Zagreb, svibanj 2019.

*Umjesto ove stranice umetnite izvornik Vašeg rada.
Da bi ste uklonili ovu stranicu obrišite naredbu \izvornik.*

Hvala svima!

SADRŽAJ

1. Uvod	1
2. Sustavi za raspodijeljenu obradu tokova podataka	2
2.1. Model objavi/pretplati	3
2.2. Apache Kafka	3
2.2.1. Objavljiivač	4
2.2.2. Tema	4
2.2.3. Posrednik	4
2.2.4. Pretinac	5
2.2.5. Pretplatnik	7
2.3. Apache Pulsar	8
2.3.1. Objavljiivac	8
2.3.2. Tema i pretplata	8
2.3.3. Posrednik	9
2.3.4. Upravljana knjiga	9
2.3.5. Pretplatnik	10
3. Unistenje višestrukih poruka	11
4. Korisnici	12
5. Iznimke	13
5.1. RabbitMQ	13
6. Postojeca rjesenja	14
7. Arhitektura rjesenja	15
8. Dizajn rjesenja	16

9. Izvedba rjesenja	17
10. Rezultati	18
11. Zaključak	19
Literatura	20

1. Uvod

Raspodijeljeni sustavi su nepouzdati. Pogreška u sklopovlju, operacijskom sustavu, programu ili mreži može izazvati ispad bilo kojeg dijela sustava. Ispadi u raspodijeljenom sustavu mogu pokrenuti lanac ispada. Ako ispad ne uzrokuje lanac ispada sustav će i dalje patiti jer će program i dalje zahtijevati računalno vrijeme i memoriju, a s njima neće obavljati koristan posao. U najgorem slučaju ispad može izazvati potpuno zatajenje sustava gdje je jedini lijek iznova pokrenuti sve njegove dijelove. U najboljem slučaju ispad će samo smanjiti učinkovitost sustava. Bez pažljivog nadzora raspodijeljenog sustava teško je otkriti ispad, a još teže otkloniti izvor ispada.

Zadatak nadzora je otkriti ispad i njegov uzrok. [9] ispade dijeli na ispad procesa, pogreške u komunikaciji, vremenske pogreške, pogrešan odgovor i bizantske pogreške. Ako se ispad želi otkriti potrebno je pratiti vrijednosti koje ukazuju da se ispad dogodio. Korisne vrijednosti mogu biti zauzeće memorije, brzina obrade zahtjeva, sadržaj poruke ili duljina uspostave komunikacijskog kanala. Nadzirani program vrijednosti predaje nadzorniku koji je čovjek ili program. Ako nadzor obavlja program on treba biti pouzdaniji od nadziranog programa inače je problem nadzora udvostručen, a ne riješen.

Prvi korak u izradi pouzdanog sustava nadzora je izrada pouzdanog sakupljača vrijednosti. Prije same izrade istražene su ideje, sukobljene su arhitekture i uspoređena postojeća rješenja. Kako bi naglasak na idejama, arhitekturama i izradi rješenja bio podjednak napravljen je sakupljač za jedan raspodijeljeni sustav. Kada bi umjesto njega bio napravljen svestran sakupljač koji može sakupljati vrijednosti raznog sklopovlja, operacijskih sustava i programa misli o arhitekturi bile bi izražene na uštrp onima o izradi programa. Nadzirani sustav je Apache Kafka, popularni sustav za raspodijeljenu obradu tokova podataka.

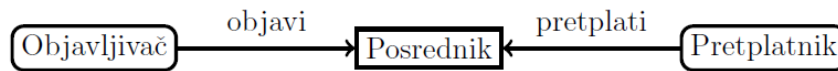
2. Sustavi za raspodijeljenu obradu tokova podataka

[3] razlikuje tok podataka od skupa podataka. Toku podaci dolaze stalno, u nepoznatom redoslijedu, bez znaka kada će prestat i bit će odbačeni nakon što su pročitani. Ovi sustavi su raspodijeljeni jer se tok mora obraditi brzo. Primjer problema toka podatka su kartično plaćanje, praćenje korisnika mrežnih stranica, posluživanje reklama i preporuka. Svi navedeni problemi imaju mnogo izvora i ponora podataka. Kako sakupiti, zabilježiti, te dostaviti podatke na ponor gdje će se obraditi su problemi za koje sustavi za raspodijeljenu obradu tokova podataka nude rješenje. Zato što su tokovi podatka raznovrsni razvijeno je mnoštvo alata za njihovu obradu. Alate razlikujemo po načinu sakupljanja podataka, po mogućnostima toka, po arhitekturi i načinu zapisivanja podataka.

Prije razvoja sakupljača vrijednosti potrebno je usporediti i izabrati alat za raspodijeljenu obradu podataka. Kako bi usporedba alata i izvedba sakupljača bila jednostavnija u obzir su uzeti samo javno dostupni alati. Većina takvih alata koristi model objavi/pretplati. Umjesto da se usporede svi javno dostupni alati izabran je predstavnik iz svakog važnog skupa alata:

- Apache Kafka je predstavnik popularnih alata za raspodijeljenu obradu tokova podataka
- Apache Pulsar je predstavnik modernih alata. [4] je 2016. objavio Pulsar dok je [6] objavio Kafku 2011. godine
- RabbitMQ je predstavnik alat koji potiskuju podatke prema korisnicima. Za razliku od RabbitMQ korisnici Kafke i Pulsara moraju povlačiti podatke

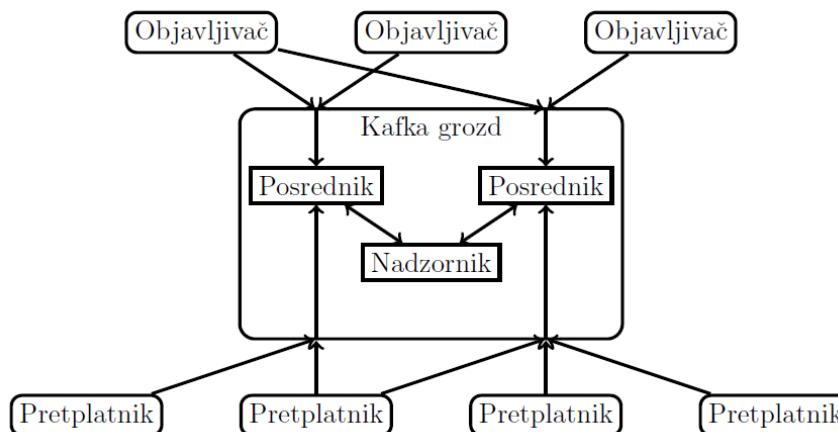
2.1. Model objavi/pretplati



Slika 2.1: Model objavi/pretplati

Apache Kafka, Apache Pulsar i RabbitMQ koriste model objavi/pretplati za razmjenu podataka između procesa. 2.1 prikazuje nužne dijelove modela objavi/pretplati. [9] navodi da se model objavi/pretplati sastoji od objavljiivača koji šalju poruke, pretplatnika koji čitaju poruke i posrednika koji razmjenjuje poruke između njih. Posrednik zapisuje objavljene poruke i dopušta pretplatnicima da ih čitaju. Prednost modela objavi/pretplati nad izravnim razgovorom među procesima je što objavljiivači i pretplatnici ne trebaju znati jedan za drugoga.

2.2. Apache Kafka



Slika 2.2: Apache Kafka grozd

Apache Kafka je popularni alat za raspodijeljenu obradu tokova podataka. 2.2 prikazuje Kafka grozd. Svaki grozd sastoji se od ZooKeeper nadzornika i barem jednog posrednika. Nadzornik usklađuje posrednike. Objavljiivači šalju poruke u temu grozda. Posrednici zapisuju poruke u pretinac teme. Pretplatnici čitaju poruke iz teme grozda i obrađuju podatke. [2] [6] opisuju primjenu, arhitekturu i izvedbu Kafka pretplatnika, pretinca, posrednika, teme i objavljiivača.

2.2.1. Objavljivač

Objavljivač je korisnički program koji šalje poruke u temu grozda. Poruka se nužno sastoji od sadržaja, teme na koju se objavljuje, oznake pretinca, odmaka u pretincu i vremena objave. Poruke se šalju u skupini nakon sto prođe određeno vrijeme ili se nakupi dovoljno poruka. [7] navodi kako poruke sažeti prije slanja. Tema je podijeljena na pretince. Pretinac je dnevnik u koji posrednik zapisuje poruke. Posrednik je poslužitelj koji zapisuje poruke u pretinac. Objavljivač ili ključem izabere pretinac u koji želi zapisati poruke ili se poruke šalju u svaki pretinac jednoliko. Posrednik će poslati potvrdu kada se poruke zapišu u pretinac vođe i izabrani broj usklađenih sljedbenika. Prije slanja sljedećeg skupa poruka objavljivač može pričekati potvrdu posrednika.

Ako se dogodi ispad objavljivača ili posrednika, objavljivač će ponovno poslati poruke. Objavljene poruke biti će dostavljene posredniku barem jednom. Ako je uvišestručavanje poruka nedopustivo, objavljivač može koristiti transakcijski način rada.

2.2.2. Tema

Tema je tok poruka, apstraktna umotvorina u koju objavljivači šalju poruke, a iz koje pretplatnici čitaju poruke. Posrednici se brinu da poruke budu zapisane. Biti pretplaćen na temu znači napraviti podtok poruka. Poruke objavljene u temu se ravnomjerno raspoređuju u podtokove. Svaki podtok ima pokazivač kojim pretplatnik čita poruke. Ako pokazivač dođe do kraja podtoka, pretplatnik se blokira dok se ne objavi nova poruka. Količina nepročitanih poruka u temi ne utječe na protok poruka kroz temu. Poruke u temi se brišu tek nakon određenog vremena. Tema je podijeljena na pretince s kojih posrednici čitaju i u koje pišu poruke. Kako bi poruke u pretincu bile dostupne nakon ispad posrednika, pretinac je moguće umnožiti.

2.2.3. Posrednik

Posrednik je poslužiteljski program koji zapisuje poruke objavljivača i poslužuje poruke pretplatnicima. Svaki posrednik dio je samo jednog Kafka grozda. Posrednici koriste Apache ZooKeeper za izvršavanje sporazumnog algoritma, pamćenje tema, njihovih pretinaca i usklađenih posrednika. Posrednik će poruke zapisati u pretinac onim redoslijedom kojim je objavljivač poslao poruke, ne redoslijedom kojim je posrednik primio poruke. Pretplatnici će s posrednika čitati poruke onim redoslijedom kojim su zapisane u pretinac.

Kada objavljivač objavi poruke na temu posrednik će zapisati poruke u određeni

pretinac. Posrednik koristi sustav straničenja i tvrdi disk za zapisivanje poruka opisano u [5]. Umjesto da posrednik koristi memoriju procesa za priručnu memoriju posrednik poruke predaje sustavu straničenja operacijskog sustava. Operacijski sustav će poruke zapisati u stranice i pohraniti u nedodijeljene dijelove radne memorije. Poruke se zapisuju na tvrdi disk samo kada operacijski sustav želi osloboditi memoriju sustava straničenja. Opisano gospodarenje memorijom dopusta posredniku da čuva veliku količinu poruka bez gubitka učinkovitosti.

Pretplatnici često dohvaćaju uzastopne poruke pa se one čitaju iz priručne memorije umjesto iz tvrdog diska. Zahvaljujući sustavu straničenja priručna memorija sa zapisanim porukama će postojati neko vrijeme nakon ispada posrednika. Korištenjem sustava straničenja za zapisivanje poruka izbjegava se korištenje sakupljača smeća Java virtualnog stroja. Umjesto da se poruke zapisu dva puta, jednom u dodijeljenu memoriju procesa i jednom u sustav straničenja poruke se zapisu samo jednom u sustav straničenja.

Kada pretplatnik želi pročitati poruke posrednik će pronaći, pročitati i poslati poruke pretplatniku. [1] [8] opisuju kako posrednik s `sendfile` i `zero-copy` funkcijama čita poruke iz radne memorije i šalje u memoriju mrežne kartice u jednom koraku. Kako bi se dodatno ubrzao rad posrednici, objavljiivači i pretplatnici grupe poruka čitaju, zapisuju i šalju u istom binarnom obliku. Prije slanja poruka pretplatnicima posrednik može sažeti poruke.

2.2.4. Pretinac

Pretinac je logicki dnevnik i najmanja gradivna jedinica teme. Kada objavljiivac objavi poruku na temu posrednik poruke zapise u pretinac teme. Dnevnik je izveden kao skup datoteka iste velicine. Kada posrednik zapise poruke u pretinac on doda poruke na kraj datoteke. Datoteke se predaju sustavu stranicenja operacijskog sustava tek nakon određenog vremena ili dodavanja poruka. Svaka poruka ima jedinstvenu oznaku koja je ujedno i odmak unutar datoteke. Odmak sljedeće poruke racuna se kao zbroj odnaka i velicine ranije poruke. Zato su jedinstvene oznake poruka strogo rastuci ali nisu uzastopni. **SLIKA-ODMAKA-I-FILEOVA**

Protok poruka kroz temu može biti toliki da se jedan posrednik preoptereti. Zato je temu moguće podijeliti na više pretinaca. Ako je B broj posrednika u ZooKeeper grozdu onda tema može biti podijeljena na najviše B pretinaca. Svaki pretinac teme mora biti dodijeljen razlicitom posredniku kko bi se povecao protok poruka.

Ako se dogodi ispad posrednika, poruke u pretincu postat će nedostupne. Ako po-

ruke stalno moraju biti dostupne pretinac se mora umnožiti. Svaki pretinac može biti umnožen najviše B puta ako je B broj posrednika u ZooKeeper grozdu. Svaki umnoženi pretinac biti će dodijeljen razlicitom posredniku jer dodijela istom ne povećava dostupnost uslijed ispada. Ako se pretinac umnoži B puta onda će poruke u pretincu postati nedostupne tek ako se dogodi ispad na više od B-1 posrednika.

Kada su pretinci umnoženi potrebno je ujednačiti poruke u svakom umnoženom pretincu. Zato će jedan posrednik biti voda pretinca, a ostali posrednici će biti pratitelji. Svaki posrednik može biti voda najviše jednog pretinca po temi i biti pratitelj najviše B-1 pretinca po temi. Voda pretinca je jedini posrednik koji smije citati i pisati poruke u pretinac. Zadatak pratitelja je preusmjeriti korisnike na vodu pretinca i uskladiti svoj pretinac s pretincem vode. Pratitelji mogu preusmjeriti korisnike na vodu pretinca tako da pitaju ZooKeepera tko je voda kojeg pretinca. Svaki pratitelj ima pretplatnika koji čita poruke iz pretinca vode i zapisuje poruke u umnoženi pretinac.

Osim što voda pretinca je jedini posrednik koji može pisati i citati poruke iz pretinca, on se mora brinuti o pratiteljima. Voda smatra da je pratitelj živ samo ako je prijavljen u ZooKeeper grozd i ako ne zaostaje s citanjem poruka iz pretinca vode. Ako zaostaje, voda će pratitelja izbaciti iz skupa uskladenih pratitelja (ISR). Kada voda primi poruke od objavljivača on će poruke zapisati u svoju particiju i čekat će potvrde pratitelja. Tek kada svi pratitelji procitaju i zapisu nove poruke u svoj umnoženi pretinac će voda pretinca poslati potvrdu objavljivaču. Pretplatitelji mogu citati samo one poruke koje su i voda i pratitelji zapisali u svoje pretince.

Ako se posredniku koji je voda pretinca dogodi ispad, nitko neće moći citati ili pisati poruke u pretinac. Zato će pratitelji glasati tko će od pratitelja postati novi voda pretinca. Samo pratitelji koji su u skupu ukladenih pratitelja mogu postati novi voda. U iznimnom slučaju kada ne postoji uskladenih pratitelja pratitelj koji nije uskladen može postati novi voda pretinca.

Posrednika, tema, pretinaca i umnoženih pretinaca je puno. Kako posrednici nebi glasali za novog vodu pretinca za svaki pretinac zasebno jedan od posrednika je zadužen da bude nadglednik glasanja. Ako se dogodi ispad posrednika, nadglednik glasanja će ubrzati glasanje novog vode pretinca.

SLIKA prikazuje tri posrednika u ZooKeeper grozdu. Tema "sales" je podijeljena na najveći mogući broj pretincana, broj posrednika u grozdu. Kako se poruke nebi izgubile zbog ispada jednog posrednika svaki pretinac teme umnožen je dva puta. Voda prvog pretinca nasumično je postao posrednik broj jedan. Voda pretinca dva će biti ili posrednik dva ili tri jer se vodstvo i pretinci moraju podijeliti jednoliko. Ako voda pretinca dva postane posrednik tri, onda će voda pretinca tri postati posrednik dva. Pra-

titelj pretinca jedan ciji je voda posrednik jedan je posrednik tri. Pratitelj pretinca dva ciji je voda posrednik tri je posrednik dva. Pratitelj pretinca tri ciji je voda posrednik dva je posrednik jedan.

2.2.5. Pretplatnik

Pretplatnik je korisnicki program koji cita poruke iz teme. Svaki pretplatnik je clan samo jedne skupine pretplatnika. Skupina pretplatnika moze se sastojati od samo jednog clana. Skupina pretplatnika zajedno cita poruke iz teme. Svaki pretplatnik u skupini pretplatnika zaduzen za citanje poruka ima barem jedan pretinac iz kojeg jedino on moze citati poruke. Skupina pretplatnika moze istovremeneno napraviti najvise P citanja ako je P broj pretinaca i u skupini pretplatnika je barem P clanova. Poruke se mogu procitati jedino iz pretinca vode. Ako pretplatnik uputi zahtjev za citanje pratitelju pretinca on ce pretplatnika preusmjeriti na vodu pretinca.

Pretplatnik u zahtjevu za citanje navodi odmak zadnje procitane poruke i koliko poruka zeli procitati. Posrednik ce dostaviti sve poruke od odmaka do trazene kolicine ili dok ne dode do kraja pretinca. Odmak poruke je i jedinstvena oznaka poruke i mjesto u datoteci posrednika gdje se poruka nalazi. Posrednik ne pazi koje je poruke pretplatnik procitao. Pretplatnik je zaduzen za rukovanje odmakom. Odmak zadnje pocitane poruke pretplatnik salje u posebnu temu posrednika. Ako se dogodi ispad pretplatnika, drugi pretplatnik ce procitati odmak iz teme i nastaviti s citanjem poruka iz pretinca.

Posrednik ce jednom pretplatniku po skupini pretplatnika dostaviti poruku barem jednom. Ako se dogodi ispad pretplatnika ista poruka se moze dostaviti vise puta. Ako je nedopustivo istu poruku procitati vise puta onda korsnik mora napraviti vlastiti algoritam ili koristiti transakcijskog pretplatnika. Pretplatnik cita poruke onim redoslijedom kojim su poruke zapisne u pretinac. Posrednici vremenski ne ureduju dostavu poruka iz svih pretinaca, ali su poruke vremenski uredeno u svakom pretincu pojedinačno. Ako je potrebno vremenski urediti sve poruke u svim pretincima onda je potrebno razviti vlastiti algortitam ili napraviti temu s samo jednim pretincem.

Vise skupina pretplatnika moze istovremeno citati poruke iz iste teme i pretinaca. Posrednik ce odaslati istu poruku svim pretplacenim pretplatnicima. Posrednici ce skupu pretplatnika omoguciti citanje poruka samo ako su svi pratitelji u skupu usklađenih pratitelja i voda pretinca zapisali poruke u pretinac. Nakon sto pretplatnik primi poruke on ce u temi odmaka azurirati odmak do kojeg je procitao poruke.

Model povlacenja poruka pretplatnicima dopusta citanje poruka brzinom koja njima

odgovara. Model dopusta i ucinkovito razasiljanje iste poruke na vise pretplatnika. Ako pretplatnik zeli ponovno procitati poruku on se moze vratiti unatrag i poslati odmak procitane poruke. Kako pretplatnik nebi zaglavio u petlji ako u pretincu nema novim poruka on sebe moze blokirati dok ne dodu nove poruke. Posrednik nemoze samostalno pobrisati poruke iz pretinca. To je zato sto posrednik nezna koje poruke su procitali svi pretplatnici. Posrednici su zato napravljeni da s povecanjem neprocitanih poruka njihova ucinkovitost ne opada. Posrednik ce poruke izbrisati nakon zadanog vremena.

2.3. Apache Pulsar

2.3.1. Objavljiivac

Objavljiivac je korisnicki program koji salje poruke u temu. Poruke se nuzno sastoje od sadrzaja, oznake objavljiivaca, jedinstvene oznake poruke i vremena objave. Poruke se mogu slati u skupini. Prije slanja, poruke se mogu sazeti. Tema moze biti podijeljena na pretince. Objavljiivac moze usmjeriti poruke u odredeni pretinac. Poruke se mogu usmjeriti na jedan nasumicni pretinac, na točno određene pretince koristeći ključ ili se mogu slati jednoliko na sve pretince. Objavljiivci ili čekaju potvrdu da je posrednik zapisao poruke ili nastave s radom i nakonadno provjeravaju jel su poruke primljene.

SLIKA GROZDA

2.3.2. Tema i pretplata

Tema je imenovani tok poruka koji služi za prijenos poruka od objavljiivaca do pretplatnika. Svaka tema je izgrađena kao poveznica. Svaka tema pripada prostoru imena. Prostor imena je upravna jedinica kojom se mijenjaju postavke skupini tema. Svaki prostor imena pripada stanaru. Stanari pripadaju Pulsar grozdu. Moguće je napraviti bezmemorijske teme koje poruke čuvaju do slanja pretplatnicima ili ispada posrednika. Bezmemorijske teme objavljene poruke potiskuju prema pretplatnicima. Prednost bezmemorijskih tema je brzina. Kolicina neprocitanih poruka u temi ne utjece na protok poruka kroz temu. Poruka se brise iz teme kada svi pretplatnici procitaju poruku ili kada je poruka procitana i starija od zadane vrijednosti ili kada je neprocitana i starija od zadane vrijednosti. Tema se moze podijeliti na vise podtema zvanih pretinac. Pretinci su ravnomjerno raspodijeljeni po posrednicima.

SLIKA RAPOSIJELE PRE-TINACA

Kako objavlјivaci usmjeravaju dostavu poruka na pretince tako pretplatnici citaju poruka koristeći pretplate. Postoje tri vrste pretplate. Isklјuciva pretplate pravo citanja poruka daje samo jednom pretplatniku. Zajednicka pretplate jednoliko dostavlja poruke pretplatnicima. Sigurnosna pretplate pravo citanja daje samo jednom pretplatniku dok se pretplatniku ne dogodi ispad. Kada se dogodi ispad pomocni pretplatnik ce nastaviti citanje poruka od mjesta ispada. Zajednicka pretplate ne podrzava skupnu potvrdu dostave poruka niti pazi na vremensko uredenje dostave poruka. SLIKA PRETPLATA

2.3.3. Posrednik

Posrednik je program bez stanja koji se sastoji od dva dijela. Prvi dio je posluzitelj s REST suceljem za upravljanje i pretrazivanje tema, a drugi dio je otpravnik za prijenos podataka. Umjesto da objavlјivaci i pretplatnici izravno razgovaraju s posrednikom mogu se spojiti na zastupnika koji ce preusmjeriti njihove zahtjeve posrednicima.

Pulsar proces satoji se od barem jednog Pulsar grozda. Pulsar grozd sastoji se od barem jednog posrednika, Apache BookKeeper grozda i Apache ZooKeeper grozda. Pulsar grozdovi mogu umoziti poruke ako pripadaju istom procesu. Zadaca posrednik je posluziti poruke pretplatnicima iz upravlјane knjige ili BookKeepera, dok je ZooKeeper zaduzen za cuvanje podata o grozdu. BookKeeper se brine o zapisivanju poruka.

2.3.4. Upravlјana knjiga

BookKeeper grozd je raspodijeljeni zapisivac koji se sastoji od barem jednog zapisnicara. Zapisnicar zapisuju poruke koje mu posrednici posalju. Poruke se mogu umoziti i zapisati u vise knjiga odjednom. Svaka tema sastoji se od barem jedne knjige. Kapacitet poruka koji se moze zapisati moze se povecati dodavanjem zapisnicara. Zapisnicari mogu istovremeno citati i pisati poruke.

Knjiga je struktura podatka u koju se poruke mogu dodati samo na kraj. Nakon sto se knjiga zatvori ona se jedino moze citati. Ako se zapisnicaru dogodi ispad, knjiga se zatvori. Kada se ispad otkloni zapisnicar ce ustanovit u kojem je stanju knjiga i ustanovljeno stanje poslati ostalim zapisnicarima u grozdu.

Upravlјana knjiga je skup BookKeeper knjiga u koju se upisuju poruke koje pripadaju jednoj temi. Makar se poruke mogu zapisati u samo jednu knjige vise BookKeeper knjiga olaksava brisanje i pisanje poruka. Upravlјana knjiga sastoji se skupa tokova podata koji se zapisuju u knjigu s jednim pisacem i skupa pokazivaca koji prate

koje poruke su pretplatnici procitali. Zapisivac prije pisanja poruke u upravljanu knjigu poruke zapisuje u dnevnik.

2.3.5. Pretplatnik

Pretplatnik je korisnicki program koji se pretplacuje na pretplatu teme. Postoje tri vrste pretplate. Svaka pretplata određuje način na koji pretplatnik čita poruke. Pretplatnik može ili biti blokiran dok posrednik ne dobije poruku ili nastaviti s radom i dobiti budućnostnicu kojom će citati poruke. Pretplatnik može pojedinačno ili skupno potvrditi poruke.

Ako korisnik nije zadovoljan izvedenim pretplatnikom on može koristiti sučelje citaca. Sučelje citaca je biblioteka koja omogućuje ručno potvrđivanje poruka, ponovno citanje poruke i odbacivanje uvisestrucenih poruka.

3. Unistenje višestrukih poruka

Poruka se može objaviti više puta. Posrednik neće zapisati poruku koju zna da je već zapisana.

4. Korisnici

Korisnik se pitati tko je zaduzen za temu. Nakon sto dobije adresu posrednika, spojit ce se na posrednika.

5. Iznimke

Koje funkcionalnosti ne rade s kojima. TODO

5.1. RabbitMQ

6. Postojeca rjesenja

7. Arhitektura rjesenja

8. Dizajn rjesenja

9. Izvedba rjesenja

10. Rezultati

11. Zaključak

Zaključak.

LITERATURA

- [1] Linux sendfile, September 2017. URL <http://man7.org/linux/man-pages/man2/sendfile.2.html>.
- [2] *Official Kafka Documentation*, 2019. URL <https://kafka.apache.org/documentation/>.
- [3] B. Babcock, S. Babu, M. Datar, R. Motwani, i J. Widom. Models and issues in data stream systems. Technical Report 2002-19, Stanford InfoLab, 2002. URL <http://ilpubs.stanford.edu:8090/535/>.
- [4] J. Francis i M. Merli. Open-sourcing pulsar, pub-sub messaging at scale, September 2016. URL <https://yahooeng.tumblr.com/post/150078336821/open-sourcing-pulsar-pub-sub-messaging-at-scale>.
- [5] P.-K. Kamp. Notes from the architect. URL <http://varnish-cache.org/docs/trunk/phk/notes.html>.
- [6] N. Kreps, J. Narkhede i J. Rao. Kafka: a distributed messaging system for log processing. June 2011. URL <https://www.microsoft.com/en-us/research/wp-content/uploads/2017/09/Kafka.pdf>. ACM 978-1-4503-0652-2/11/06; NetDB workshop 2011.
- [7] N. Narkhede. Kafka compression, February 2017. URL <https://cwiki.apache.org/confluence/display/KAFKA/Compression>.
- [8] S. Palaniappan i P. Nagaraja. Linux sendfile, September 2008. URL <https://developer.ibm.com/articles/j-zero-copy/>.
- [9] I. P. Žarko, K. Pripužić, I. Lovrek, i M. Kušek. *Raspodijeljeni sustavi*, 1.3 izdanju, 2013.

**Stvarnovremensko praćenje parametara ispravnosti rada u sustavu za
raspodijeljenu obradu tokova podataka**

Sažetak

Sažetak na hrvatskom jeziku.

Ključne riječi: Ključne riječi, odvojene zarezima.

Real-Time Health Monitoring in Distributed Data Stream Processing System

Abstract

Abstract.

Keywords: Keywords.