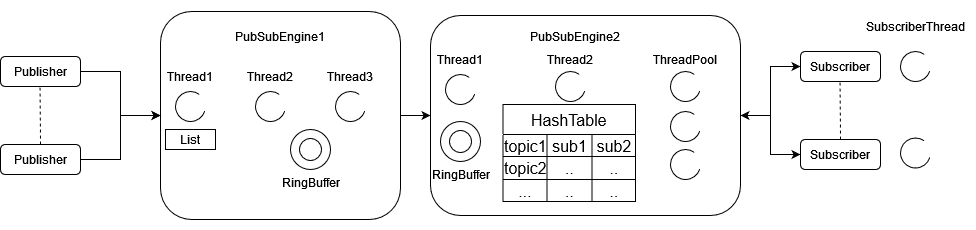
IKP Projekat 8

1. Uvod

Potrebno je napraviti dva PubSub servisa koji mogu da opslužuju proizvoljan broj klijenata. Servisi treba medjusobno da komuniciraju tako da jedan servis samo prima poruke od Publisher-a i zatim pošalje poruke drugom PubSub servisu. Drugi servis prima poruke od prvog PubSub servisa i zatim poruke prosledjuje Subscriber-ima.

2. Dizajn



Publisher – konzolna aplikacija koja se konektuje, kreira poruke i šalje ih na PubSubEngine.

Subscriber - konzolna aplikacija koja se konektuje na PubSubEngine i vrši prijavljivanje na topic za koji želi da dobija vesti. Omogućava biranje više topica pomoću thread-a.

PubSubEngine1 – konzolna aplikacija koja služi za komunikaciju sa publisher-om i za prosleđivanje poruka na PubSubEngine2. Sadrži niti:

Thread1 – prihvata konekcije od publisher-a i smešta ih u listu.

Thread2 – provera događaje nad listom prihvaćenih socket-a i preuzete poruke smešta u ring buffer. U slučaju prekida konekcije briše socket iz liste.

Thread3 – prosleđuje poruke iz ring buffer-a u PubSubEngine2

PubSubEngine2 – konzolna aplikacija koja služi za komunikaciju sa subscriberima, prihvata poruke od PubSubEngine1. Sadrži niti:

Thread1 – prihvata poruke i smešta ih u ring buffer.

Thread2 – prihvata konekcije od subscriber-a i smešta ih u listu prihvaćenih socket-a. Proverava događaje nad prihvaćenim socket-ima i u slučaju pretplate na novi topic dodaje socket u hash tabelu, a u slučaju prekida uklanja ga iz liste i hash tabele.

ThreadPool (Threads 3, 4, 5) – preuzima poruke iz ring buffer-a i na osnovu topic-a pronalazi u hash tabeli pretplaćene subscriber-e i prosleđuje im poruke.

3. Struktura podataka

Od struktura podataka implementirani su List, HashTable i RingBuffer.

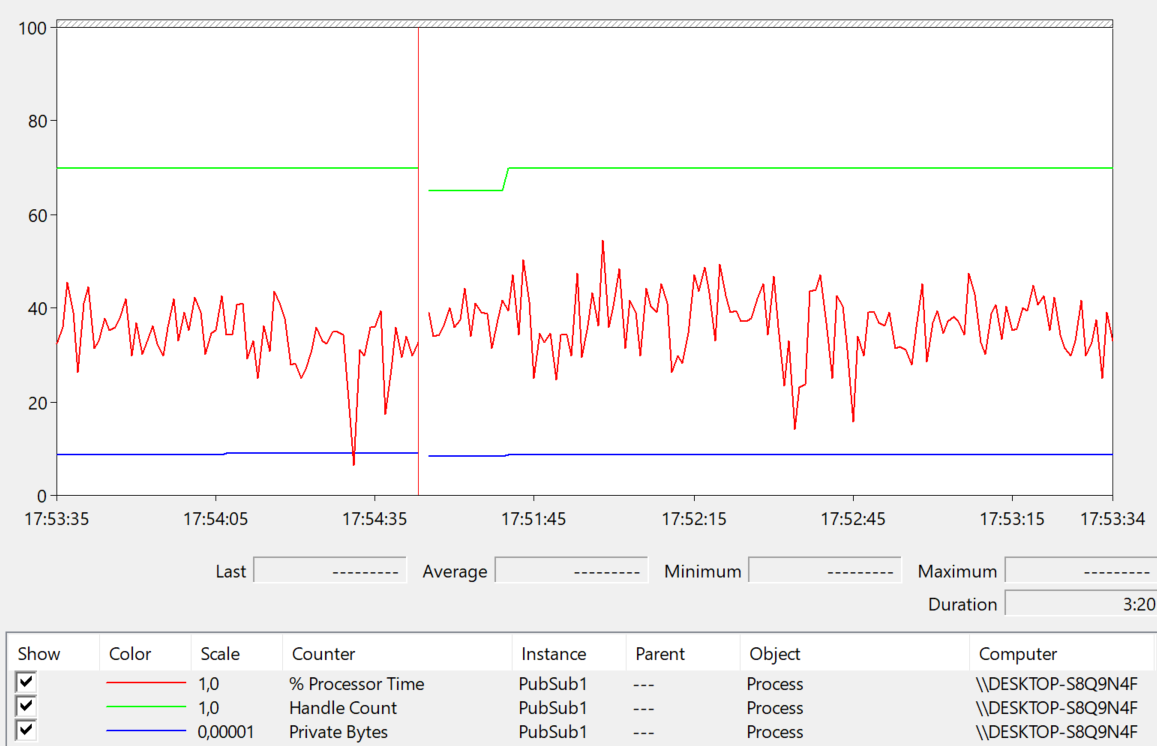
List – sadrži poseban tip za publisher-a i subscriber-a. Čuva socket-e na koji su konektovani, a za subscriber-e sadrži polje topic na koji su pretplaćeni. Omogućava pretragu, dodavanje i brisanje uz dinamičku alokaciju memorije.

HashTable - Predstavlja kolekciju koja sadrži strukturu tipa Subscribers koja se sastoji od topic-a i List-e prijavljenih subscriber-a na taj topic. Ova struktura je izabrana jer je optimalno rešenje za dobavljanje List-e subscriber-a prijavljenih na traženi topic.

RingBuffer – FIFO struktura čiji je zadatak privremeno čuvanje pristiglih podataka strukture tipa PORUKA, koja sadrži topic i text. Ova struktura izabrana je zbog najoptimalnijeg korišćenja memorije. Sadrži operacije za dodavanje i uzimanje sa ringbuffer-a.

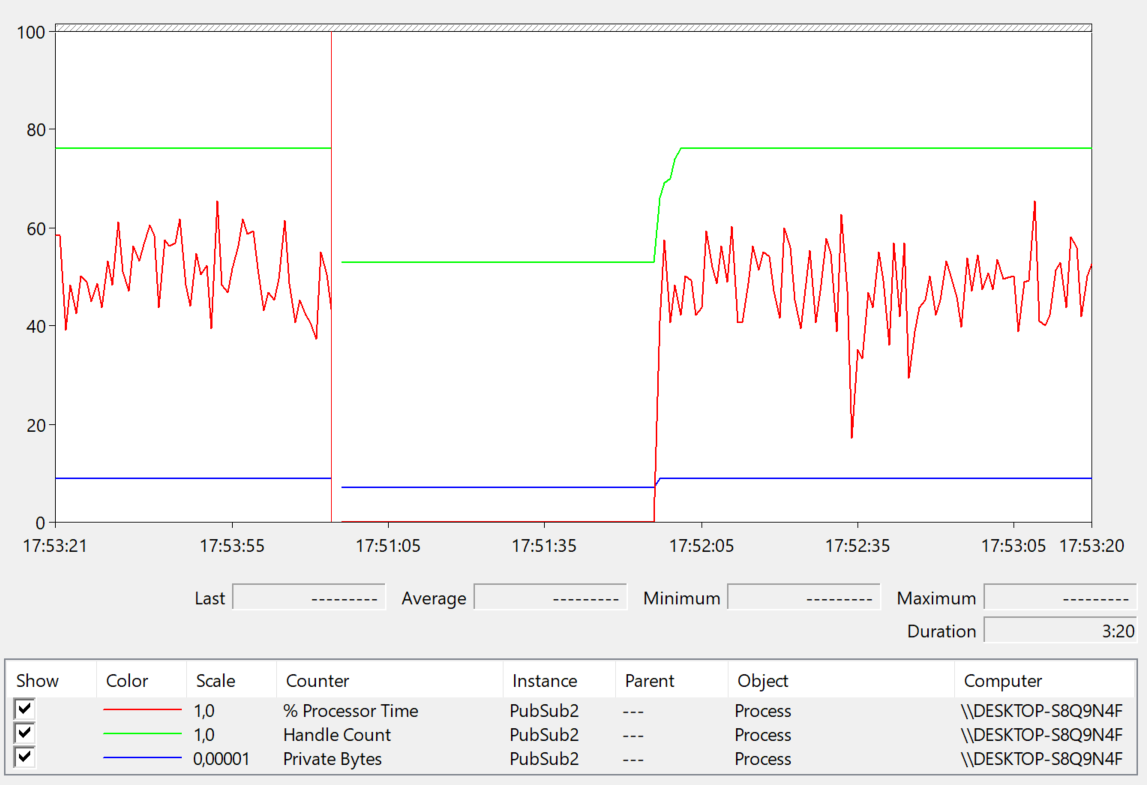
4. Struktura podataka

Prvi stres test – Publisher šalje 500 kratkih poruka naizmenično za dva različita topic-a. Posmatra se PubSubEngine1.

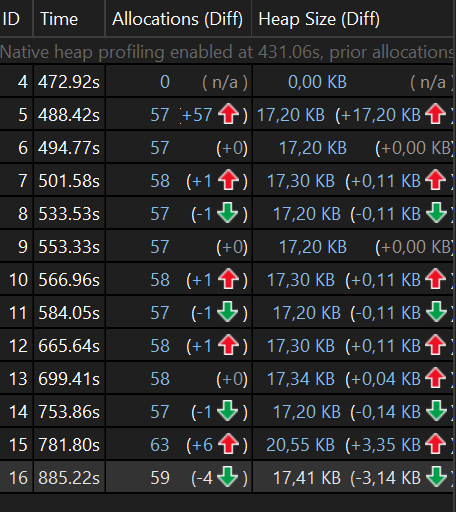


Slika 1. Performance monitor grafika PubSub1

Drugi stres test - Publisher šalje 500 kratkih poruka naizmenično za dva različita topic-a. Posmatra se PubSubEngine2.



Slika 2. Performance monitor grafika PubSub2



Slika 3. Heap u toku stres testova

5. Zaključak

Na osnovu izgleda heap-a na slici 3 može se zaključiti da ne dolazi do curenja memorije. U oba testa opterećenje procesora je očekivano. Broj HANDLE-ova se vraća na istu poziciju.

6. Potencijalna unapređenja

Sistem se može unaprediti poboljšavanjem komunikacije između dva PubSubEngine-a dodavanjem više komunikacionih kanala ili obostranim korišćenjem deljene memorije umesto tcp kanala. Takođe se može povećati broj thread-ova u thread pool-u za efikasniju isporuku poruka.