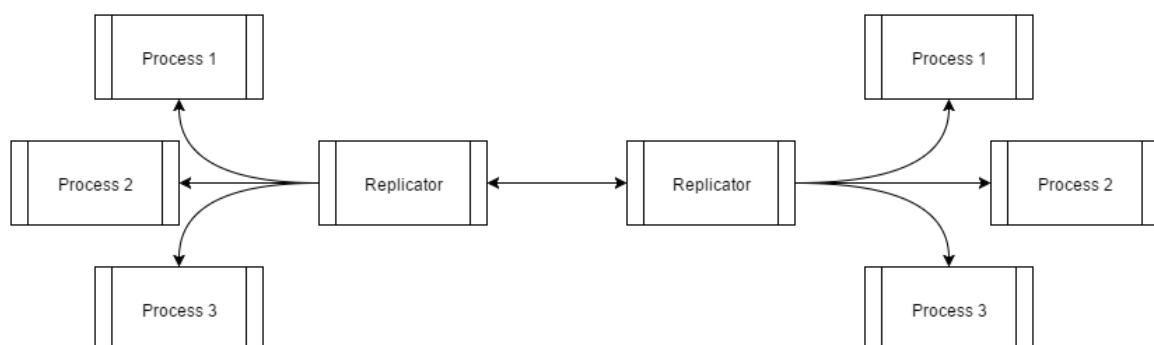


Projekat 15 – Replikator



Asistent : Bojan Jelačić

Radili: PR66-2019 Stefan Šćekić i PR34-2019 Nemanja Radovanović

Contents

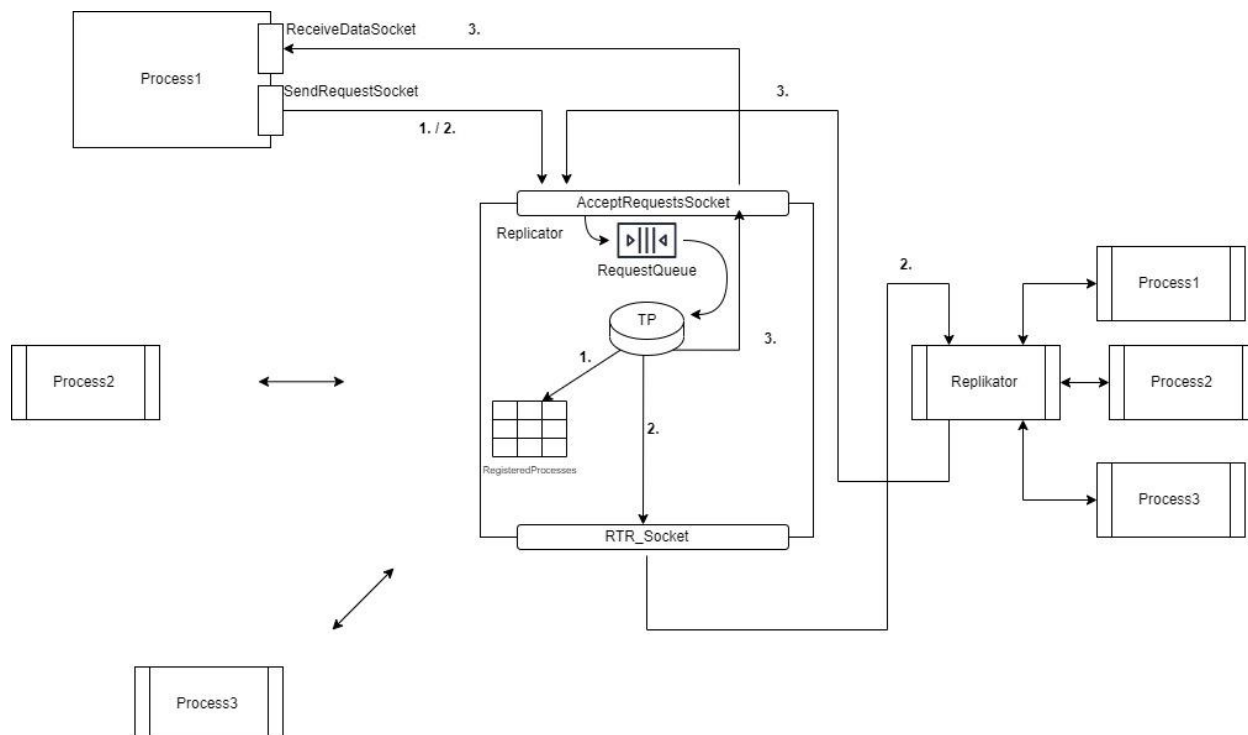
Uvod.....	3
Dizajn.....	4
Strukture podataka	6
Rezultati testiranja	7
Zaključak	8
Potencijalna unapređenja	9

Uvod

Ovaj projekat pokušava da reši problem replikacije podataka sa jedne lokacije na drugu tako što će pružiti servis na koji procesi mogu da se priključe.

Naš cilj pri izradi projekta je bio da se postigne što veća dostupnost servisa, kako bi najveći broj procesa mogao biti priključen, kao i što brža usluga servisa i najbolje iskorišćenje sistemskih resursa.

Dizajn



Slika prikazuje preporučenu arhitekturu servisa sa po jednim otvorenim replikatorom i procesom radi objašnjena rada servisa.

Proces se sastoji od 2 soketa, SendRequestSocket služi za slanje zahteva replikatoru, dok ReceiveDataSocket služi za primanje podataka za replikaciju.

Svi zahtevi se šalju u strukturi request koja će biti objašnjena u poglavlju Strukture Podataka.

Pokretanje procesa se može izvršiti pozivom:

```
.\Process.exe [client_id] [server_port] [mode]
```

client_id : id koji će biti dodeljen pokrenutom procesu

server_port : port replikatora na koji će klijent probati da se konektuje

mode : mod u kom će klijent biti pokrenut, 0 je za ručno upravljanje, 1 za test registracije,

2 za test replikacije

Replikator se sastoji od AcceptRequestSoketa, RTRSoketa, RequestQueue-a, ThreadPool-a i HashTabele registrovanih procesa.

AcceptRequestSocket : služi za primanje zahteva od procesa i drugog replikatora

RTRSocket : služi za slanje podataka na replikaciju drugom replikatoru

RequestQueue : služi za primanje zahteva od AcceptRequestSocket-a

ThreadPool : služi za preuzimanje i obradu zahteva iz RequestQueue-a

HashTable : služi za pamćenje registrovanih procesa kako bi im se kasnije mogli poslati podatci

Replikator se može pokrenuti sa sledećim pozivom:

```
.\Replicator.exe [server_port] [replicator_port]
```

Server_port : port na kojem će AcceptRequestSocket biti otvoren

replicator_port : port drugog replikaotra na koji će replikator probati da se poveže

Tok podataka 1. prikazuje registraciju procesa.

Tok podataka 2. prikazuje slanje podataka na replikaciju.

Tok podataka 3. prikazuje primanje podataka za replikaciju.

Strukture podataka

Ovaj projekat koristi 5 struktura podataka.

1. Dynamic Queue – Koristi se za skladištenje zahteva, odabran je zbog FIFO načina rada i promenjive veličine.
2. HashTable sa uvezanom listom – Koristi se za skladištenje informacija o procesu, specifično za skladištenje njegovog id-a i soketa sa kojim je povezan na replikator. Odabran je kako bi se smanjilo vreme potrebno za pretragu.
3. Custom struktura Request – sastoji se od RequestCoda (enumeracija koja sadrži tipove requestova koji mogu da se šalju, odabran je zbog usklađenja tipova zahteva izmedju procesa i replikatora) i podataka koji će biti poslani.
4. Custom struktura Process – sastoji se od id-a i soketa koji mu pripada, koristi se za skladištenje podataka o procesu u HashTabeli.
5. Custom struktura File – sadrži podatke potrebne za replikaciju fajla, ti podatci su owner_id, file_name, file_contents i file_length. Koristi se za prenos fajla preko mreže.

Rezultati testiranja

Trenutni datum i vreme je 22.1.2023 19:38, zbog kasnog kucanja ove dokumentacije ovaj deo će biti preskočen ali testovi koji su proveravali oslobađanje memorija korišćene od strane promenljivih i thread-ova, kao i iskorišćenost procesora su bili izvođeni tokom izrade projekta i mogu se ponoviti na odbrani.

Projekat dolazi uz 2 skripte za testiranje i grupom test podataka za replikaciju.

Zaključak

Testovi koji su sprovedeni tokom izrade projekta su dali očekivane rezultate, sva zauzeta memorija je bila i oslobođena i korišćenje semafora za usklađenje rada niti je dalo očekivane rezultate čuvanja procesorskog vremena(ne radi na 100% tokom rada programa).

Potencijalna unapređenja

Unapređenja se ne odnose toliko na optimizaciju performansi koliko na rad samog programa, ali moguća unapređenja su sledeća:

Dodavanje menadžera resursima, koji će dinamički menjati zauzete resurse replikatora u zavisnosti od količine zahteva na čekanju.

Dodavanje zaštite od repliciranja bilo čega na host računar.

Dodavanje mogućnosti rada na različitim računarima.

I još dosta toga, slobodno pitajte na odbrani, imam još 15min da predam projekat.