




https://github.com/monberin/microservices_architecture/tree/hw2

Members 

Search 


Member							Owned ...	OS Total...	OS Com...	OS Free...	OS CPU...	OS Max ...
127.0.0.1:5701		Disa	Disa	Enal	No	5	91	7.65 GB	7.52 GB	179.12 MB	8.00 %	524288
127.0.0.1:5702		Disa	Disa	Enal	No	5	90	7.65 GB	7.52 GB	180.41 MB	4.96 %	524288
127.0.0.1:5703		Disa	Disa	Enal	No	5	90	7.65 GB	7.52 GB	179.40 MB	7.74 %	524288

1 - 3 of 3 Rows 10 


Продемонструйте роботу Distributed Map

- використовуючи API створіть Distributed Map
- запишіть в неї 1000 значень з ключем від 0 до 1к
- подивитись як зміниться розподіл якщо відключити одну ноду/дві ноди. Чи буде втрата даних?


3 ноди:

Map Statistics (In-Memory Format: BINARY) Now 


Member	Entries	Gets	Puts	Removals	Sets	Entry M...	Backups	Backup ...
127.0.0.1:57...	343	0	0	0	343	44.52 kB	326	42.30 kB
127.0.0.1:57...	335	0	0	0	335	43.47 kB	345	44.78 kB
127.0.0.1:57...	322	0	0	0	323	41.78 kB	329	42.70 kB
TOTAL	1000	0	0	0	1001	129.78 kB	1000	129.78 kB

1 - 3 of 3 Rows 10 


2 ноди:

Map Statistics (In-Memory Format: BINARY) Now 


Member	Entries	Gets	Puts	Removals	Sets	Entry M...	Backups	Backup ...
127.0.0.1:57...	473	0	0	0	335	61.39 kB	527	68.39 kB
127.0.0.1:57...	527	0	0	0	323	68.39 kB	473	61.39 kB
TOTAL	1000	0	0	0	658	129.78 kB	1000	129.78 kB

1 - 2 of 2 Rows 10 

1 нода:

Map Statistics (In-Memory Format: BINARY) Now 

Member	Entries	Gets	Puts	Removals	Sets	Entry M...	Backups	Backup ...
127.0.0.1:57...	1000	0	0	0	323	129.78 kB	0	0 B
TOTAL	1000	0	0	0	323	129.78 kB	0	0 B

1 - 1 of 1 Rows 10 

Втрати даних немає, дані кожного разу діляться +- симетрично

Продемонструйте роботу Distributed Map with locks

- використовуючи 3 підключення (чи підключення з 3х клієнтів) одночасно запустіть приклади за посиланням з документації з підрахунку значень в циклі: а) без блокування; б) з песимістичним; в) з оптимістичним блокуванням
- порівняйте результати кожного з запусків

```
map_wo_locks ×  
/home/monberin/Documents/archited  
starting...  
time taken: 11.592589855194092  
result: 1000
```

```
map_pessimistic ×  
/home/monberin/Documents/archited  
starting...  
time taken: 12.42386507987976  
result: 1000
```

```
map_optimistic ×  
/home/monberin/Documents/archited  
starting...  
time taken: 11.53001618385315  
result: 1000
```

Map Browser

Key

1

Key Type

String

Value:	1000	Class:	java.lang.Integer
Memory Cost:	64.00 B	Creation Time:	N/A
Expiration Time:	N/A	Hits:	-1
Last Access Time:	N/A	Last Update Time:	N/A
Last Stored Time:	N/A	Version:	1000
Time to Live:	292471208 years and 8 months	Max Idle:	Unlimited
Key Owner Member:	127.0.0.1:5703		

Cancel

BROWSE

Race condition-у не спостерігається навіть при версії без локів, проте необхідно уникати таких ситуацій.

Варіант з оптимістичним блокуванням найшвидший.

Налаштуйте Bounded queue

- на основі Distributed Queue налаштуйте Bounded queue
- з однієї ноди (клієнта) йде запис, а на двох інших читання
- перевірте яка буде поведінка на запис якщо відсутнє читання, і черга заповнена
- як будуть вичитуватись значення з черги якщо є декілька читачів

Queues

Search






Name	Items	Backups	Max Age	Min Age	Average Age
bounded_queue	10	10	0ms	N/A	0ms

1 - 1 of 1 Rows

10



  ~/Documents/architecture/hw2_hazelcast  python3 ./writer.py




running




writing...

no place left

no place left

ended writing

  ~/Documents/architecture/hw2_hazelcast 

  ~/Documents/architecture/hw2_hazelcast  python3 ./reader.py

reading...

taken 13_value

taken 15_value

taken 17_value

taken 19_value

taken 21_value

taken 23_value

taken 25_value

taken 27_value

taken 29_value

taken 31_value

taken 33_value

taken 35_value

taken 37_value

taken 39_value

```
~/Documents/architecture/hw2_hazelcast python3 ./reader.py
reading...
taken 1_value
taken 2_value
taken 3_value
taken 4_value
taken 5_value
taken 6_value
taken 7_value
taken 8_value
taken 9_value
taken 10_value
taken 11_value
taken 12_value
taken 14_value
taken 16_value
```

Якщо йде запис, але не відбувається читання, то дані записуватись не будуть. Якщо декілька читачів, то вони вичитують різні значення, не повторюючись.