任务一

准备编译环境

安装必须依赖

本机环境为
安装所需的依赖，如 `gcc` 、 `make` 、 `autoconf` 等。

```bash
sudo apt-get update
sudo apt-get install -y build-essential autoconf
zip unzip libx11-dev libxext-dev libxrender-dev
libxtst-dev libcups2-dev libasound2-dev ccache
libfontconfig1-dev libfreetype6-dev
```

安装 Boot JDK

编译 JDK 需要一个现有的 JDK 作为引导 JDK（boot JDK）。

```
sudo apt-get install -y openjdk-17-jdk
```

安装后使用 java -version 验证

```
No VM guests are running outdated hypervisor (qemu) binaries on this host.
root@iZbp161wkz6lckluuph7c7Z:/java# java -version
openjdk version "17.0.12" 2024-07-16
OpenJDK Runtime Environment (build 17.0.12+7-Ubuntu-1ubuntu222.04)
OpenJDK 64-Bit Server VM (build 17.0.12+7-Ubuntu-1ubuntu222.04, mixed mode, sharing)
root@iZbp161wkz6lckluuph7c7Z:/java#
```

编译 Tencent JDK

clone 后配置编译选项

在源码目录中，运行 `configure` 脚本来配置编译选项。启用 Shenandoah GC 可以通过指定相关的选项：

```
bash configure --with-jvm-features=shenandoahgc --with-boot-jdk=/usr/lib/jvm/java-17-openjdk-amd64`
```

- `--with-jvm-features=shenandoahgc`：启用 Shenandoah GC。
- `--with-boot-jdk`：指定引导 JDK 的路径，确保使用已安装的 JDK 路径。

```
configure: error: Could not find all X11 headers (shape.h Xrender.h Xrandr.h XTest.h Intrinsic.h). You might be able to fix this by running 'sudo apt-get install
libx11-dev libxext-dev libxrender-dev libxrandr-dev libxtst-dev libxt-dev'.
```

报错，按照提示重新安装

```
A new configuration has been successfully created in
/java/TencentKona-17/build/linux-x86_64-server-release
using configure arguments '--with-jvm-features=shenandoahgc --with-boot-jdk=/usr/lib/jvm/java-17-openjdk-amd64'.

Configuration summary:
* Name:            linux-x86_64-server-release
* Debug level:     release
* HS debug level: product
* JVM variants:    server
* JVM features:    server: 'cds compiler1 compiler2 epsilongc g1gc jfr jni-check jvmci jvmti management nmt parallelgc serialgc services shenandoahgc vm-structs zgc'
* OpenJDK target: OS: linux, CPU architecture: x86, address length: 64
* Version string: 17.0.12-internal+0-adhoc.root.TencentKona-17 (17.0.12-internal)

Tools summary:
* Boot JDK:       openjdk version "17.0.12" 2024-07-16 OpenJDK Runtime Environment (build 17.0.12+7-Ubuntu-1ubuntu222.04) OpenJDK 64-Bit Server VM (build 17.0.12+7-Ubuntu-1u
buntu222.04, mixed mode, sharing) (at /usr/lib/jvm/java-17-openjdk-amd64)
* Toolchain:      gcc (GNU Compiler Collection)
* C Compiler:     Version 11.4.0 (at /usr/bin/gcc)
* C++ Compiler:   Version 11.4.0 (at /usr/bin/g++)

Build performance summary:
* Cores to use:   4
* Memory limit:   7266 MB
```

成功配置 shenandoahgc，且观察发现配置为 4 核 8G

开始编译

```
root@iZbp161wkz6lckluuph7c7Z:/java/TencentKona-17# make images
```

编译成功

```
Creating jdk.security.auth.jmod
Creating jdk.security.jgss.jmod
Creating jdk.unsupported.jmod
Creating jdk.unsupported.desktop.jmod
Creating jdk.xml.dom.jmod
Creating jdk.zipfs.jmod
Compiling 3 files for BUILD_DEMO_CodePointIM
Updating support/demos/image/jfc/CodePointIM/src.zip
Compiling 3 files for BUILD_DEMO_FileChooserDemo
Updating support/demos/image/jfc/FileChooserDemo/src.zip
Creating interim jimage
Compiling 29 files for BUILD_DEMO_SwingSet2
Updating support/demos/image/jfc/SwingSet2/src.zip
Compiling 3 files for BUILD_DEMO_Font2DTest
Updating support/demos/image/jfc/Font2DTest/src.zip
Compiling 64 files for BUILD_DEMO_J2Ddemo
Updating support/demos/image/jfc/J2Ddemo/src.zip
Compiling 15 files for BUILD_DEMO_Metalworks
Updating support/demos/image/jfc/Metalworks/src.zip
Compiling 2 files for BUILD_DEMO_Notepad
Updating support/demos/image/jfc/Notepad/src.zip
Compiling 5 files for BUILD_DEMO_Stylepad
Updating support/demos/image/jfc/Stylepad/src.zip
Compiling 5 files for BUILD_DEMO_SampleTree
Updating support/demos/image/jfc/SampleTree/src.zip
Compiling 8 files for BUILD_DEMO_TableExample
Updating support/demos/image/jfc/TableExample/src.zip
Compiling 1 files for BUILD_DEMO_TransparentRuler
Updating support/demos/image/jfc/TransparentRuler/src.zip
Compiling 2 files for CLASSLIST_JAR
Creating support/classlist.jar
Creating support/demos/image/jfc/CodePointIM/CodePointIM.jar
Creating support/demos/image/jfc/FileChooserDemo/FileChooserDemo.jar
Creating support/demos/image/jfc/SwingSet2/SwingSet2.jar
Creating jdk.jlink.jmod
Creating support/demos/image/jfc/Font2DTest/Font2DTest.jar
Creating support/demos/image/jfc/J2Ddemo/J2Ddemo.jar
Creating support/demos/image/jfc/Metalworks/Metalworks.jar
Creating support/demos/image/jfc/Notepad/Notepad.jar
Creating java.base.jmod
Creating support/demos/image/jfc/Stylepad/Stylepad.jar
Creating support/demos/image/jfc/SampleTree/SampleTree.jar
Creating support/demos/image/jfc/TableExample/TableExample.jar
Creating support/demos/image/jfc/TransparentRuler/TransparentRuler.jar
Creating jdk image
Creating CDS archive for jdk image
Creating CDS-NOCOOPS archive for jdk image
Stopping sjavac server
Finished building target 'images' in configuration 'linux-x86_64-server-release'
```

编译后的目录

验证

进入 `build/linux-x86_64-server-release` 目录，查看是否有 `images/jdk` 子目录，里面应该有完整的 JDK 发行版，包括 `bin`，`lib` 等子目录。

```
root@iZbp161wkz6lckluuph7c7Z:/java/TencentKona-17# cd build/linux-x86_64-server-release
ls images/jdk
bin  conf  demo  include  jmods  legal  lib  man  release
```

可以看到基础是 OpenJDK，构建版本来源于 TencentJDK

```
root@iZbp161wkz6lckluuph7c7Z:/java/TencentKona-17/build/linux-x86_64-server-release# ./images/jdk/bin/java -version
openjdk version "17.0.12-internal" 2024-07-16
OpenJDK Runtime Environment (build 17.0.12-internal+0-adhoc.root.TencentKona-17)
OpenJDK 64-Bit Server VM (build 17.0.12-internal+0-adhoc.root.TencentKona-17, mixed mode, sharing)
root@iZbp161wkz6lckluuph7c7Z:/java/TencentKona-17/build/linux-x86_64-server-release#
```

验证**Shenandoah GC**：

```
root@iZbp161wkz6lckluuph7c7Z:/java/TencentKona-17/build/linux-x86_64-server-release# ./images/jdk/bin/java -XX:+UseShenandoahGC -XX:+PrintFlagsFinal -version | grep UseShenandoahGC
     bool UseShenandoahGC                          = true                                {product} {command line}
openjdk version "17.0.12-internal" 2024-07-16
OpenJDK Runtime Environment (build 17.0.12-internal+0-adhoc.root.TencentKona-17)
OpenJDK 64-Bit Server VM (build 17.0.12-internal+0-adhoc.root.TencentKona-17, mixed mode, sharing)
```

结果为 true，成功启用

测试

将自编译 JDK 加入到环境变量中

```
echo 'export PATH=/java/TencentKona-17/build/linux-x86_64-server-release/jdk/bin:$PATH' >> ~/.bashrc
source ~/.bashrc
```

```
root@iZbp161wkz6lckluuph7c7Z:/java/TencentKona-17/build/linux-x86_64-server-release/jdk/bin# java --version
openjdk 17.0.12-internal 2024-07-16
OpenJDK Runtime Environment (build 17.0.12-internal+0-adhoc.root.TencentKona-17)
OpenJDK 64-Bit Server VM (build 17.0.12-internal+0-adhoc.root.TencentKona-17, mixed mode)
root@iZbp161wkz6lckluuph7c7Z:/java/TencentKona-17/build/linux-x86_64-server-release/jdk/bin#
```

验证成功

上传 GCTest

```
import java.util.ArrayList;
import java.util.List;

public class GCTest {
```

```
    public static void main(String[] args) {
        List<byte[]> objects = new ArrayList<>();

        // 无限循环，持续创建对象
        while (true) {
            // 每次创建一个10MB的对象
            objects.add(new byte[10 * 1024 * 1024]);

            // 控制对象数量，防止过快OOM
            if (objects.size() > 1000) {
                objects.subList(0, 500).clear();
            }

            try {
                // 模拟一些计算，避免空转
                Thread.sleep(50);
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
        }
    }
}
```

编译程序

```
root@iZbp161wkz61ckluuph7c7Z:/java# javac GCTest.java
```

估测

假设虚拟机内存为 512M

年轻代与老年代的比例: 默认比例通常为 1:2，意味着年轻代占 1/3（约 170MB），老年代占 2/3（约 340MB）

假设新生代的 Eden 区占用 80% 的年轻代空间
建立 10 个对象左右就会触发一次 Minor GC

实际测试

命令行：

```
java -Xmx512m -XX:+UseG1GC -Xlog:gc*:g1gc.log -
Xlog:gc+phases=debug:file=gc.log:tags,uptime,time,level
,tags:filecount=1,filesize=0 GCTest

java -Xmx512m -XX:+UseZGC -Xlog:gc*:zgc.log -
Xlog:gc+phases=debug:file=gc.log:tags,uptime,time,level
,tags:filecount=1,filesize=0 GCTest

java -Xmx512m -XX:+UseShenandoahGC -Xlog:shengc*:gc.log
-
Xlog:gc+phases=debug:file=shengc.log:tags,uptime,time,l
evel GCTest

java -Xmx512m -XX:+UseSerialGC -Xlog:gc*:serialgc.log
GCTest

java -Xmx512m -XX:+UseParallelGC -
Xlog:gc*:parrallgc.log GCTest
```

```
root@iZbp161wkz6lckluuph7c7Z:/java# java -Xmx512m -XX:+UseShenandoahGC -Xlog:gc*:gc.log -Xlog:gc+phases=debug:file=gc.log:tags,uptime,time,level GCTest
Exception in thread "main" java.lang.OutOfMemoryError: Java heap space
```

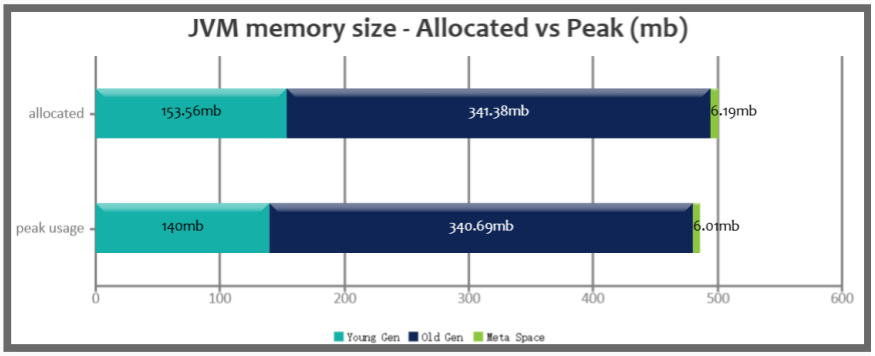
都会 OOM
得到三个 GC 的日志文件，下载下来分析

Serial

JVM memory size

(To learn about JVM Memory, click here)

Generation	Allocated	Peak
Young Generation	153.56 mb	140 mb
Old Generation	341.38 mb	340.69 mb
Meta Space	6.19 mb	6.01 mb
Young + Old + Meta space	500.19 mb	486.01 mb



年轻代和老年代占比符合预期

总体时间:

3 sec 779 ms

Total GC stats

Total GC count	13
Total reclaimed bytes	15 mb
Total GC time	784 ms
Avg GC time	60.3 ms
GC avg time std dev	49.8 ms
GC min/max time	0 / 160 ms
GC Interval avg time	268 ms

GC 总时间：784ms

Key Performance Indicators

(Important section of the report. To learn more about KPIs, click here)

1 Throughput : 79.265%

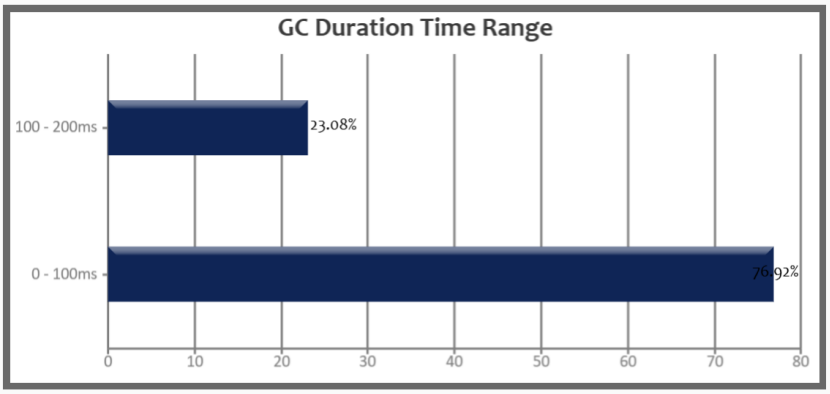
2 CPU Time : 770 ms

3 Latency:

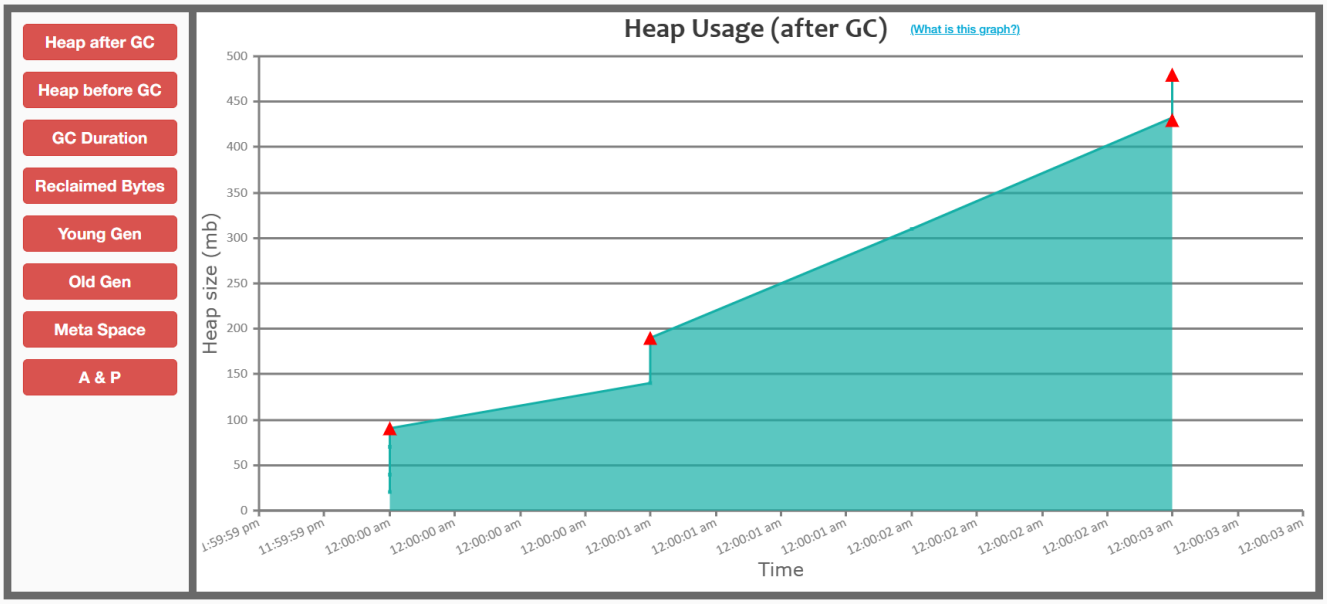
Avg Pause GC Time	60.3 ms
Max Pause GC Time	160 ms

GC Pause Duration Time Range :

Duration (ms) 100 ms Change	No. of GCs	Percentage
0 - 100	10	76.92%
100 - 200	3	23.08%



吞吐量接近 80%，平均暂停时间和最大暂停时间都较长

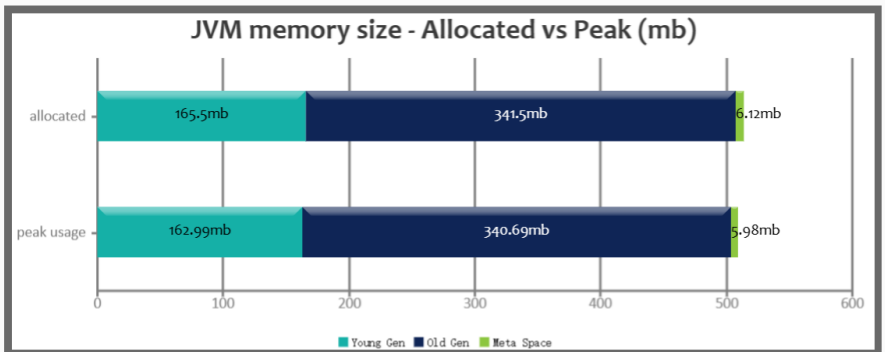


整体内存使用情况，梯度上升

Parallel

(To learn about JVM Memory, click here)

Generation	Allocated	Peak
Young Generation	165.5 mb	162.99 mb
Old Generation	341.5 mb	340.69 mb
Meta Space	6.12 mb	5.98 mb
Young + Old + Meta space	512.12 mb	508.98 mb



内存差别不大

总体时间：

3 sec 829 ms

Total GC stats

Total GC count	11
Total reclaimed bytes	n/a
Total GC time	720 ms
Avg GC time	65.5 ms
GC avg time std dev	44.2 ms
GC min/max time	10.0 ms / 140 ms
GC Interval avg time	326 ms

GC 总时间 720ms

1 Throughput : 81.196%

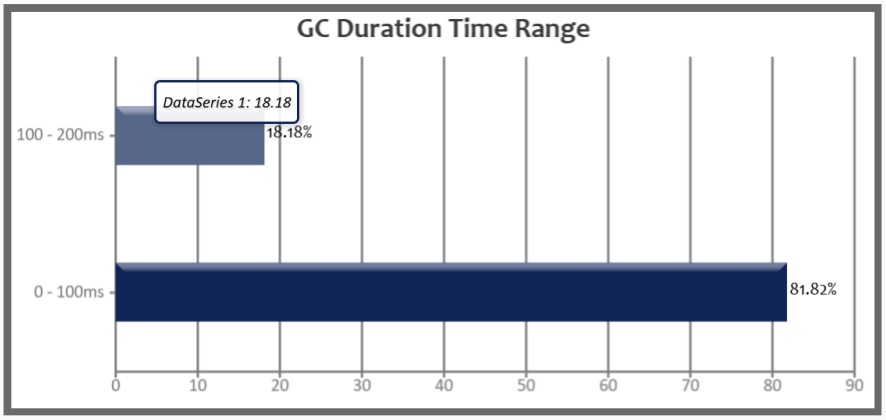
2 CPU Time : 2 sec 290 ms

3 Latency:

Avg Pause GC Time	65.5 ms
Max Pause GC Time	140 ms

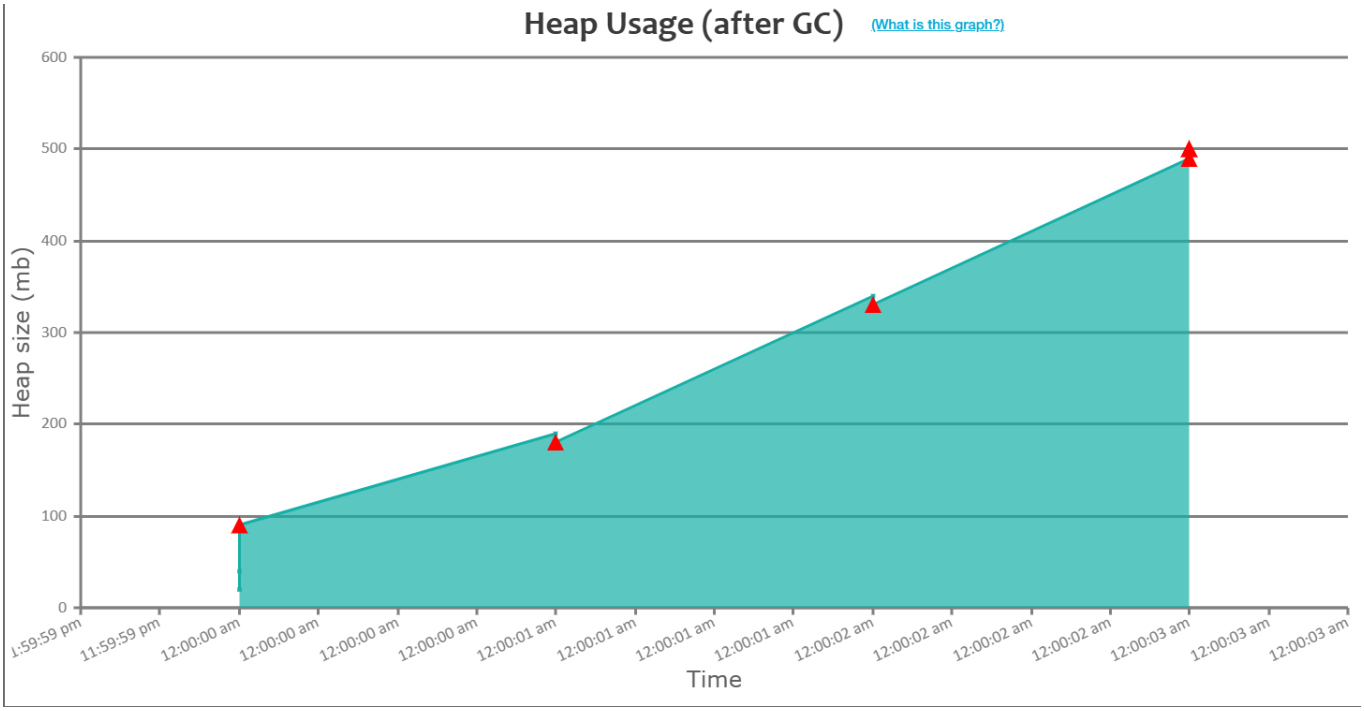
GC Pause Duration Time Range :

Duration (ms) 100 ms Change	No. of GCs	Percentage
0 - 100	9	81.82%
100 - 200	2	18.18%



吞吐量比串行略大

最大 gc 暂停时长有所下降，整体时间相对更快



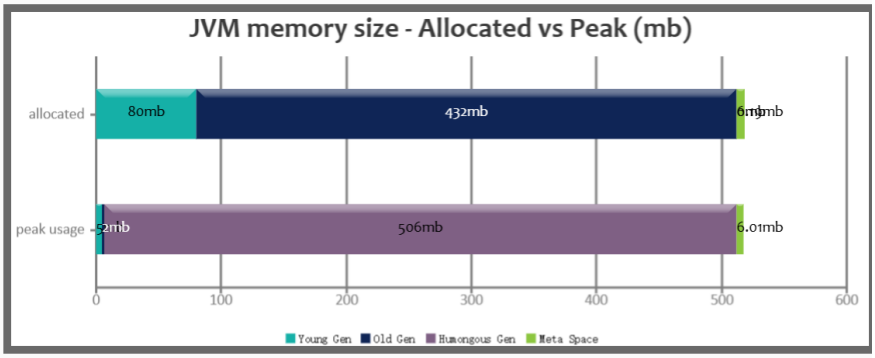
内存使用情况上升较缓

G1

```
[2024-08-28T23:20:57.007+0800][0.004s][info][gc] Using G1
[2024-08-28T23:20:57.009+0800][0.006s][info][gc,init] Version: 17.0.12-internal+0-adhoc.root.TencentKona-17 (release)
[2024-08-28T23:20:57.009+0800][0.006s][info][gc,init] CPUs: 4 total, 4 available
[2024-08-28T23:20:57.009+0800][0.006s][info][gc,init] Memory: 7266M
[2024-08-28T23:20:57.009+0800][0.006s][info][gc,init] Large Page Support: Disabled
[2024-08-28T23:20:57.009+0800][0.006s][info][gc,init] NUMA Support: Disabled
[2024-08-28T23:20:57.009+0800][0.006s][info][gc,init] Compressed Oops: Enabled (32-bit)
[2024-08-28T23:20:57.009+0800][0.006s][info][gc,init] Heap Region Size: 1M
[2024-08-28T23:20:57.009+0800][0.006s][info][gc,init] Heap Min Capacity: 8M
[2024-08-28T23:20:57.009+0800][0.006s][info][gc,init] Heap Initial Capacity: 114M
[2024-08-28T23:20:57.009+0800][0.006s][info][gc,init] Heap Max Capacity: 512M
[2024-08-28T23:20:57.009+0800][0.006s][info][gc,init] Pre-touch: Disabled
[2024-08-28T23:20:57.009+0800][0.006s][info][gc,init] Parallel Workers: 4
[2024-08-28T23:20:57.009+0800][0.006s][info][gc,init] Concurrent Workers: 1
[2024-08-28T23:20:57.009+0800][0.006s][info][gc,init] Concurrent Refinement Workers: 4
[2024-08-28T23:20:57.009+0800][0.006s][info][gc,init] Periodic GC: Disabled
[2024-08-28T23:20:57.009+0800][0.006s][info][gc,metaspace] CDS archive(s) not mapped
[2024-08-28T23:20:57.009+0800][0.006s][info][gc,metaspace] Compressed class space mapped at: 0x0000000100000000-0x0000000140000000, reserved size:
1073741824
```

最后日志中断，OOM

Generation	Allocated	Peak
Young Generation	80 mb	5 mb
Old Generation	432 mb	2 mb
Humongous	n/a	506 mb
Meta Space	6.19 mb	6.01 mb
Young + Old + Meta space	518.19 mb	512.01 mb



年轻代与老年代比例符合预期

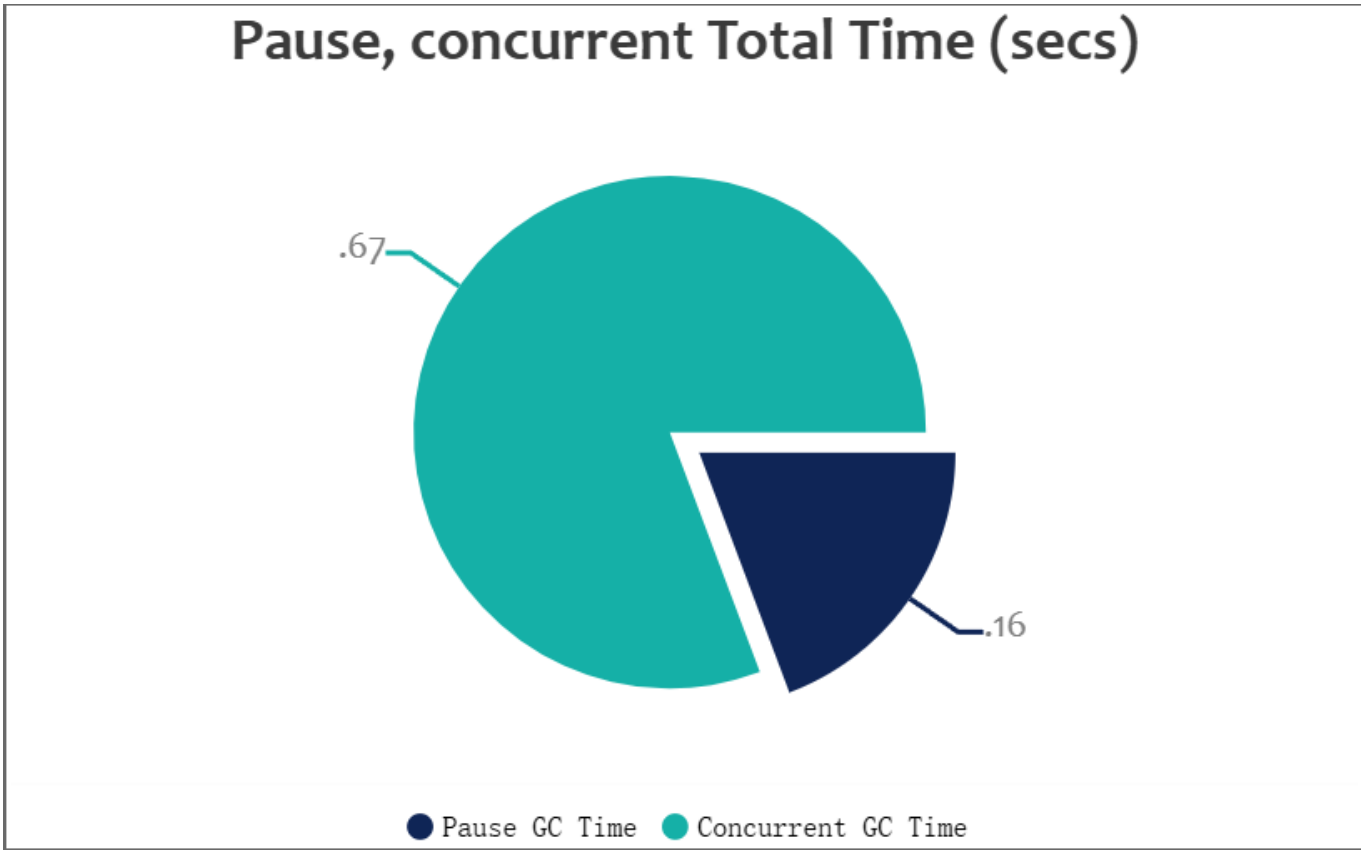
总体时间：

3 sec 173 ms

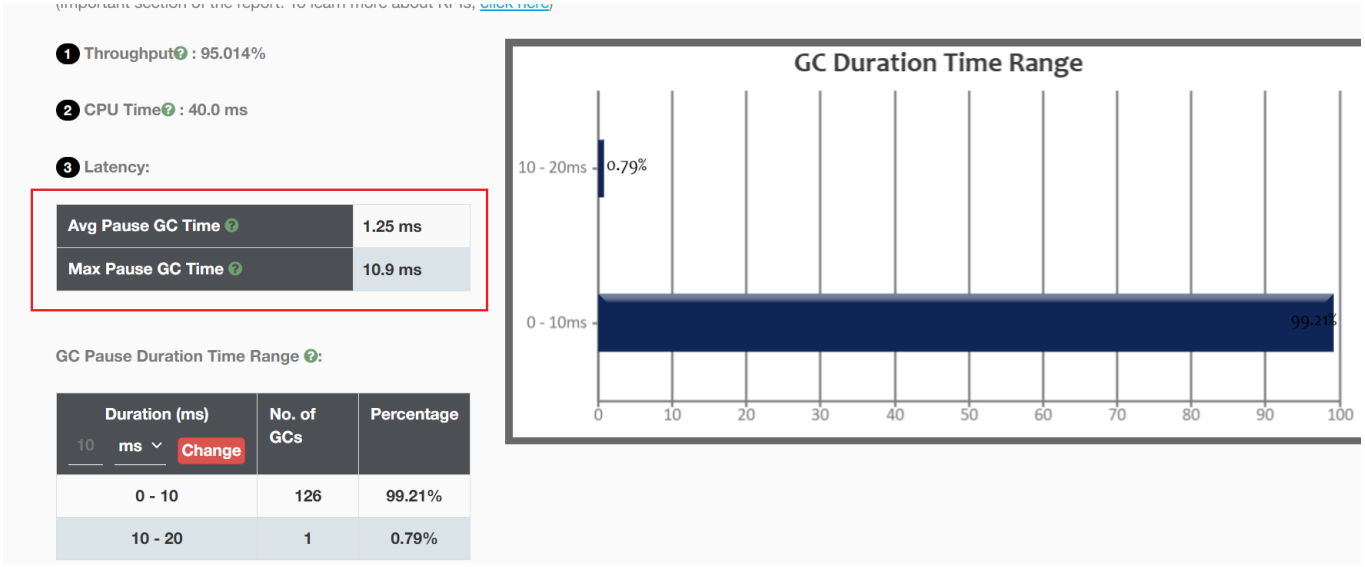
Pause Time

Total Time	**158 ms**
Avg Time	**1.25 ms**
Std Dev Time	**2.31 ms**
Min Time	**0.00700 ms**
Max Time	**10.9 ms**

stw 时间 158ms，相对短



并发的回收占比很高



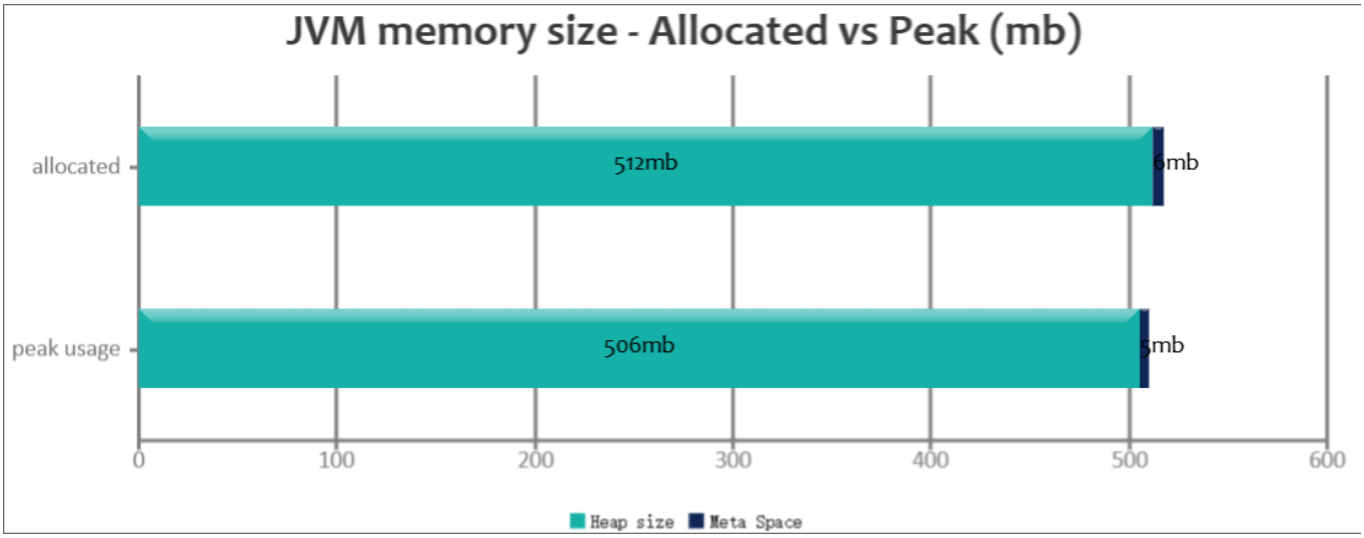
平均时间较为优异，保持 10ms

ZGC

```
[2024-08-28T23:23:43.123+0800][0.006s][info][gc,init] Initializing The Z Garbage Collector
```

```
(0%)              -                 -
[2024-08-28T23:23:46.143+0800][3.026s][info ][gc          ] GC(7) Garbage Collection (Allocation Stall) 506M(99%)->506M(99%)
[2024-08-28T23:23:46.143+0800][3.026s][info ][gc          ] Allocation Stall (main) 5.485ms
[2024-08-28T23:23:46.143+0800][3.026s][info ][gc          ] Out Of Memory (main)
[2024-08-28T23:23:46.146+0800][3.029s][info ][gc,heap,exit] Heap
[2024-08-28T23:23:46.146+0800][3.029s][info ][gc,heap,exit]  ZHeap           used 508M, capacity 512M, max capacity 512M
[2024-08-28T23:23:46.146+0800][3.029s][info ][gc,heap,exit]  Metaspace       used 6124K, committed 6272K, reserved 1114112K
[2024-08-28T23:23:46.146+0800][3.029s][info ][gc,heap,exit]   class space    used 536K, committed 640K, reserved 1048576K
```

触发了 OOM



内存利用更大

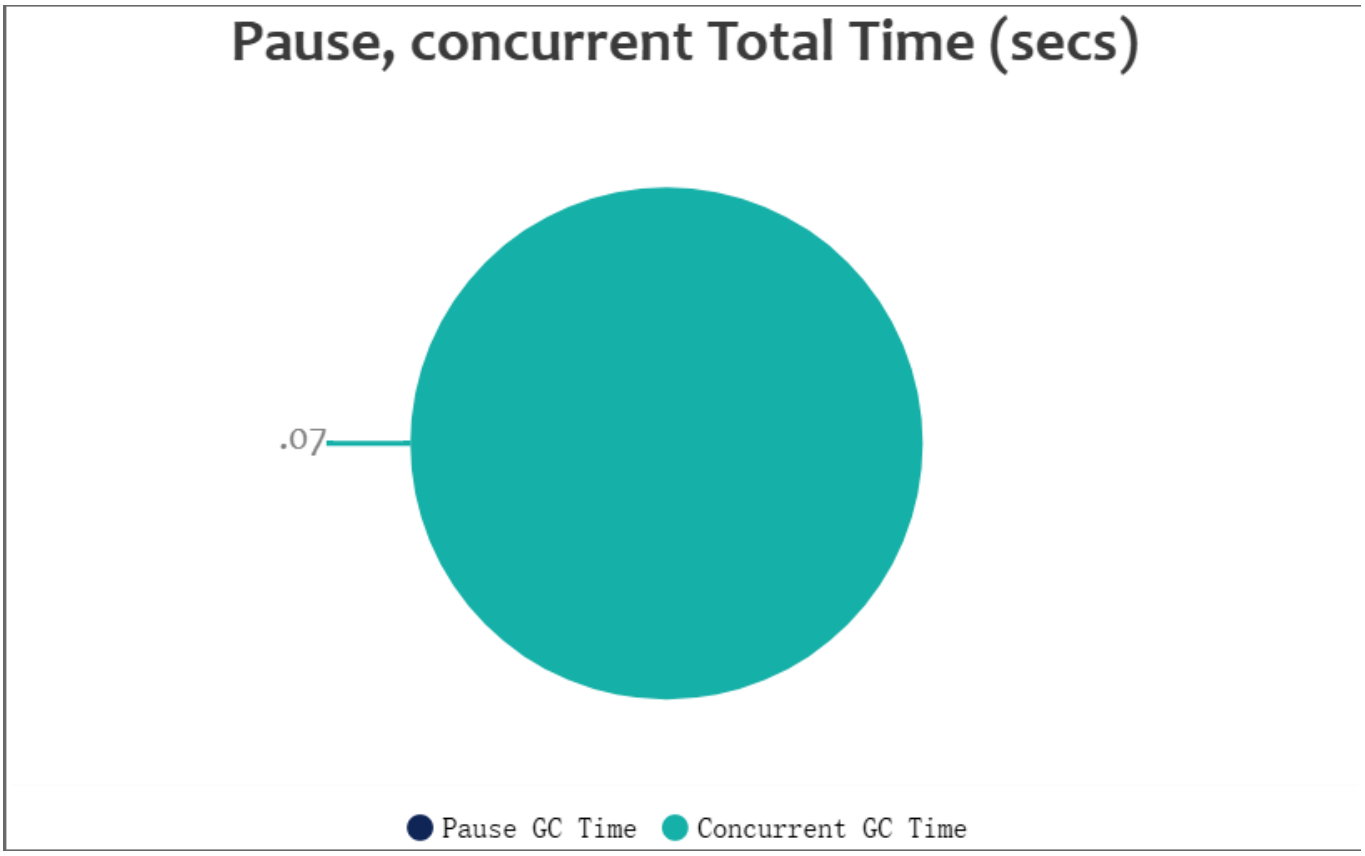
总时间：

3 sec 101 ms

Pause Time

Total Time	**0.196 ms**
Avg Time	**0.00933 ms**
Std Dev Time	**0.00345 ms**
Min Time	**0.00500 ms**
Max Time	**0.0200 ms**

极低的暂停时间



几乎全部是并发

Key Performance Indicators

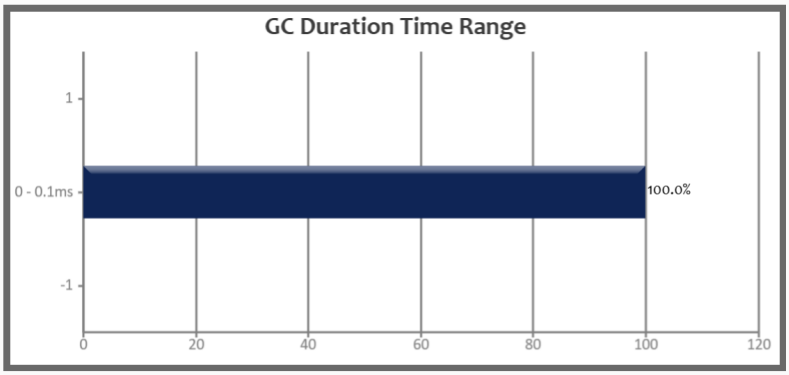
(Important section of the report. To learn more about KPIs, click here)

1. Throughput : 99.994%
2. CPU Time : n/a
3. Latency:

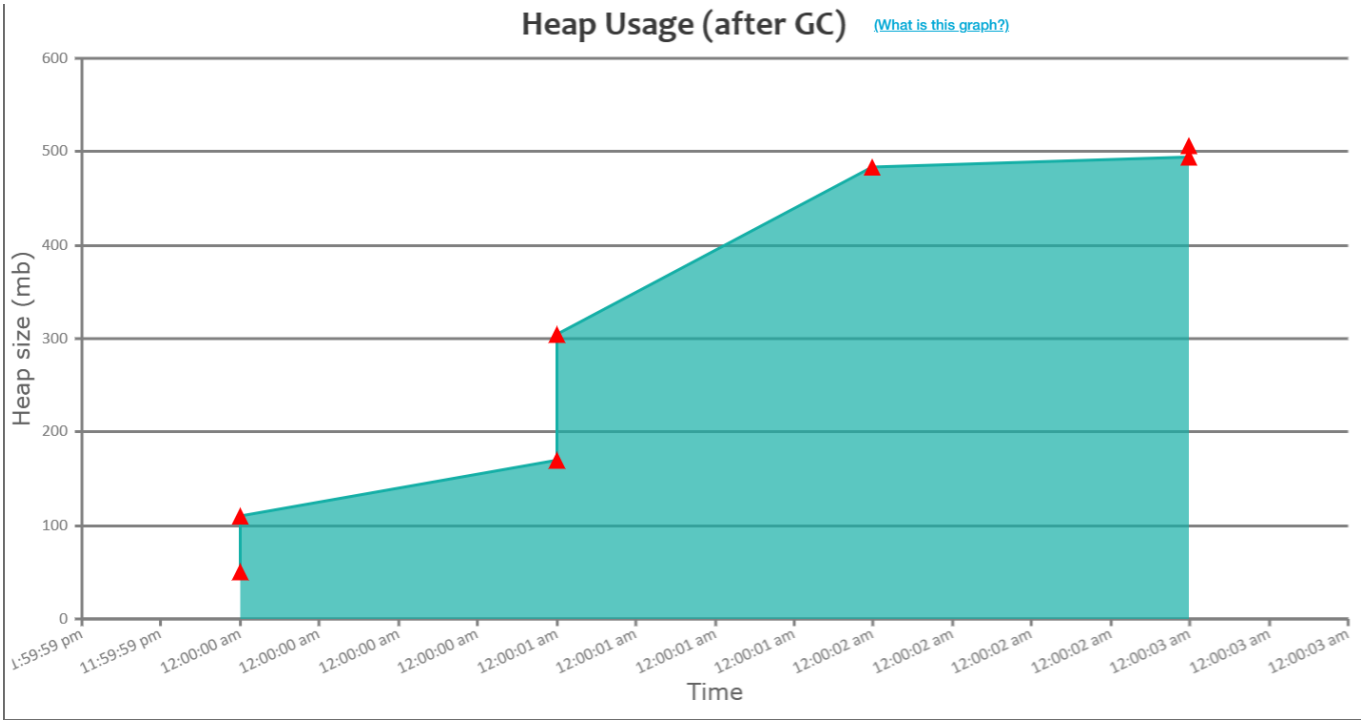
Avg Pause GC Time	0.00933 ms
Max Pause GC Time	0.0200 ms

GC Pause Duration Time Range :

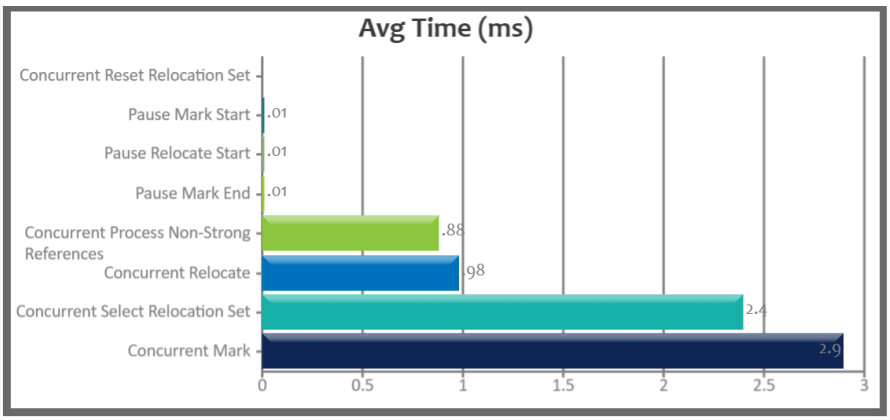
Duration (ms) 0.1 ms Change	No. of GCs phases	Percentage
0 - 0.1	21	100.0%

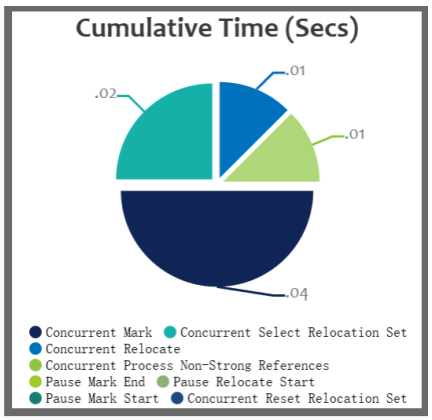


吞吐量非常高，stw 时间很短



堆内存使用阶梯式上升



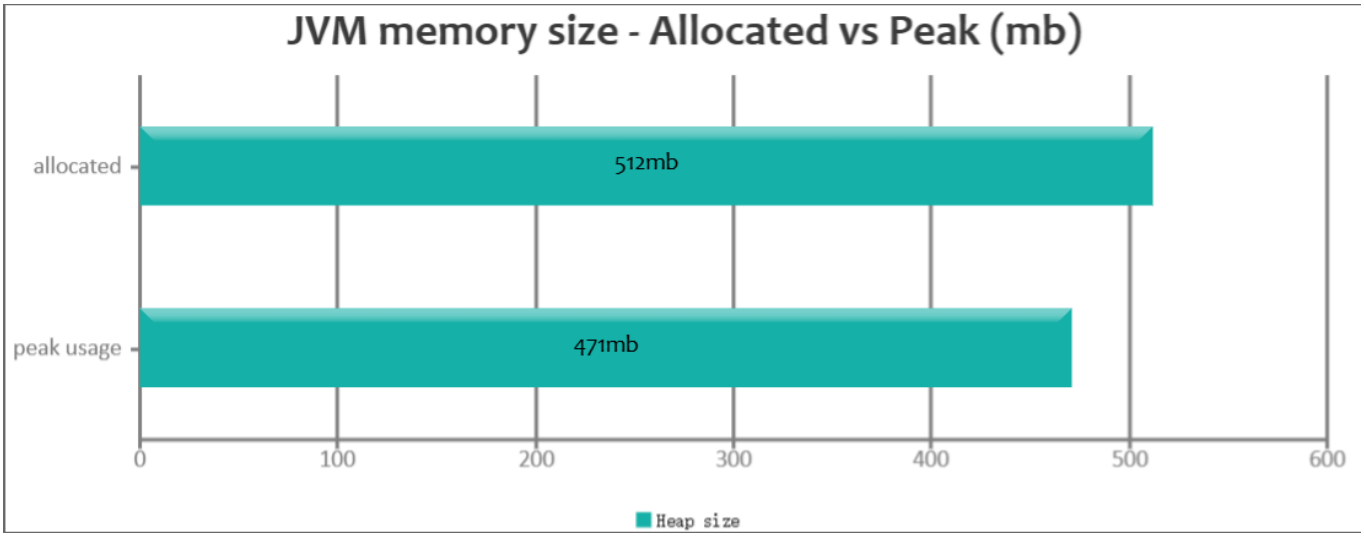


	Concurrent Mark	Concurrent Select Relocation Set	Concurrent Relocate	Concurrent Process Non-Strong References	Pause Mark End	Pause Relocate Start	Pause Mark Start	Concurrent Reset Relocation Set
Total Time	40.6 ms	16.8 ms	6.87 ms	6.19 ms	0.0890 ms	0.0540 ms	0.0530 ms	0.0100 ms
Avg Time	2.90 ms	2.40 ms	0.981 ms	0.885 ms	0.0127 ms	0.00771 ms	0.00757 ms	0.00143 ms
Std Dev Time	5.83 ms	1.44 ms	0.650 ms	0.107 ms	0.00361 ms	0.00158 ms	0.00176 ms	0.000728 ms
Min Time	0.00100 ms	1.43 ms	0.0260 ms	0.774 ms	0.00900 ms	0.00600 ms	0.00500 ms	0
Max Time	23.3 ms	5.82 ms	1.88 ms	1.02 ms	0.0200 ms	0.0100 ms	0.0100 ms	0.00200 ms
Count	14	7	7	7	7	7	7	7

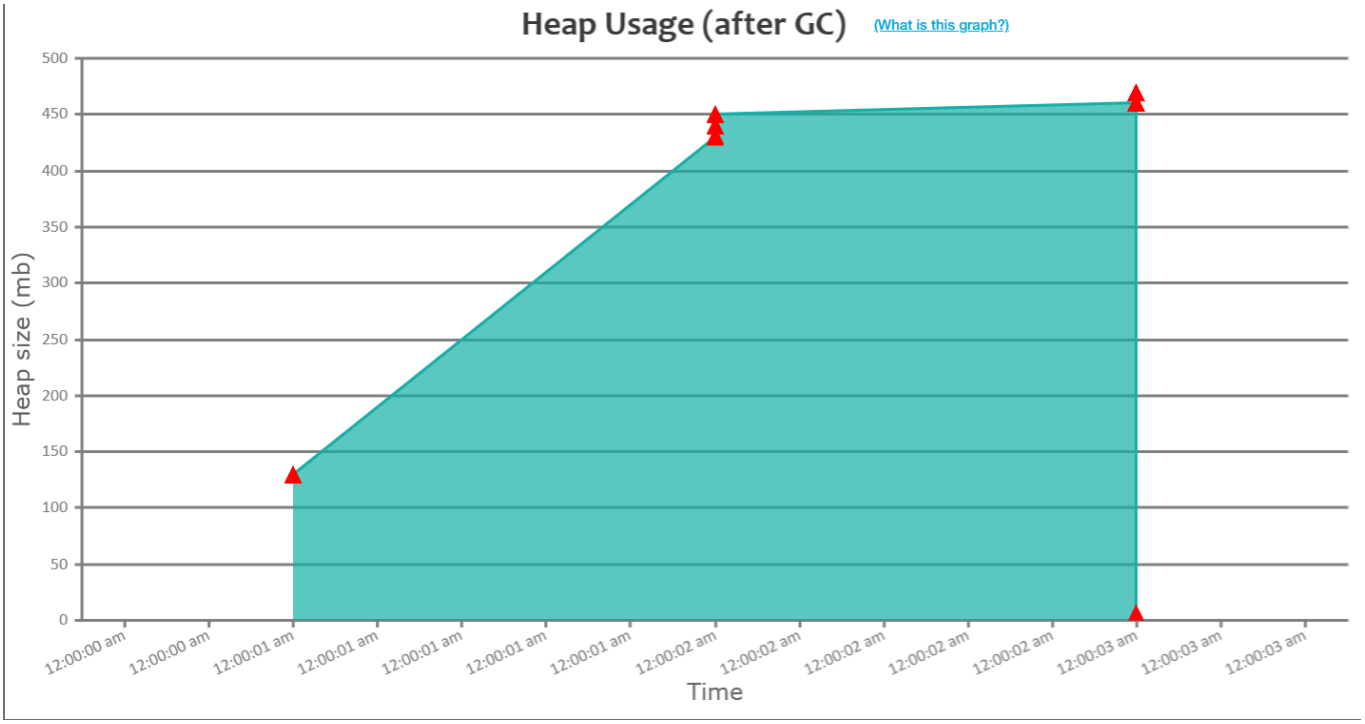
shennadoah

```
3733    [3.577s][info][gc,stats       ]
3734    [3.577s][info][gc,stats       ]
3735    [3.577s][info][gc,stats       ] Under allocation pressure, concurrent cycles may cancel, and either continue cycle
3736    [3.577s][info][gc,stats       ] under stop-the-world pause or result in stop-the-world Full GC. Increase heap size,
3737    [3.577s][info][gc,stats       ] tune GC heuristics, set more aggressive pacing delay, or lower allocation rate
3738    [3.577s][info][gc,stats       ] to avoid Degenerated and Full GC cycles.
3739    [3.577s][info][gc,stats       ]
3740    [3.577s][info][gc,stats       ]    24 successful concurrent GCs
3741    [3.577s][info][gc,stats       ]       0 invoked explicitly
3742    [3.577s][info][gc,stats       ]       0 invoked implicitly
3743    [3.577s][info][gc,stats       ]
3744    [3.577s][info][gc,stats       ]     5 Degenerated GCs
3745    [3.577s][info][gc,stats       ]       5 caused by allocation failure
3746    [3.577s][info][gc,stats       ]         5 happened at Outside of Cycle
3747    [3.577s][info][gc,stats       ]       5 upgraded to Full GC
3748    [3.577s][info][gc,stats       ]
3749    [3.577s][info][gc,stats       ]     5 Full GCs
3750    [3.577s][info][gc,stats       ]       0 invoked explicitly
3751    [3.577s][info][gc,stats       ]       0 invoked implicitly
3752    [3.577s][info][gc,stats       ]       0 caused by allocation failure
3753    [3.577s][info][gc,stats       ]       5 upgraded from Degenerated GC
3754    [3.577s][info][gc,stats       ]
3755    [3.577s][info][gc,stats       ]
3756
```

分析



最大利用为 471mb



内存利用情况

总时间:
3 sec 573 ms

Pause Time

Total Time	451 ms
Avg Time	4.51 ms
Std Dev Time	36.2 ms
Min Time	0.0190 ms
Max Time	364 ms

总暂停时间 451ms

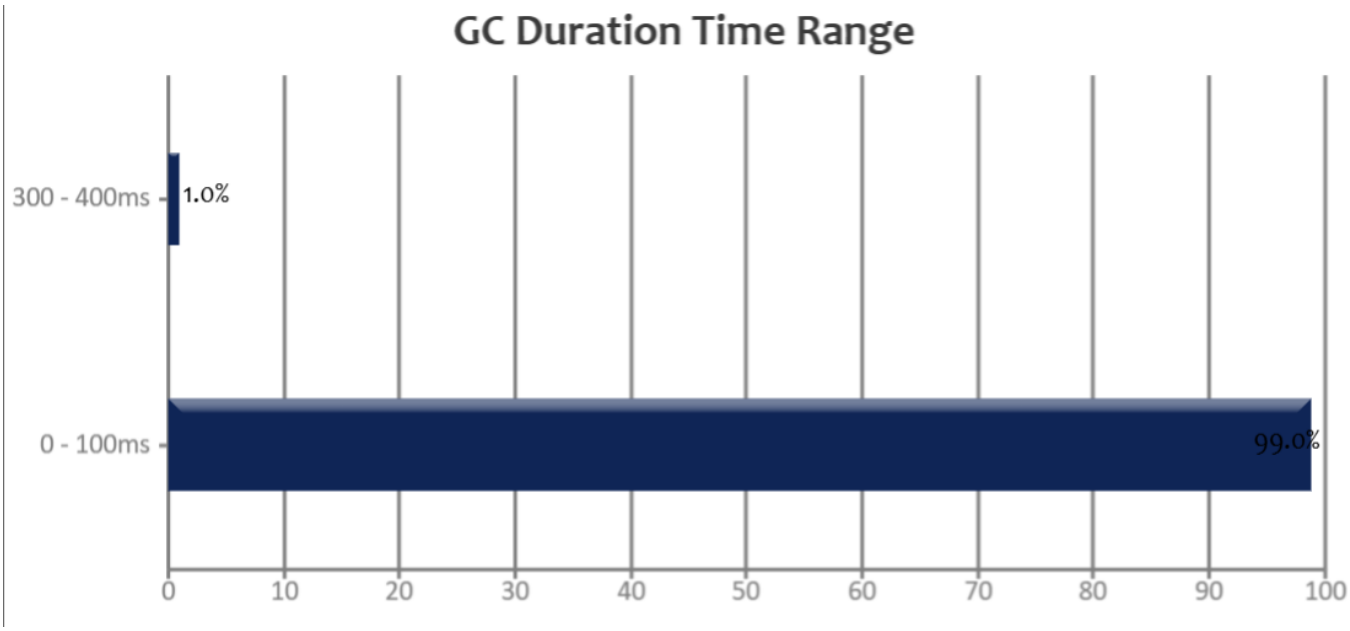
1 Throughput : 87.369%

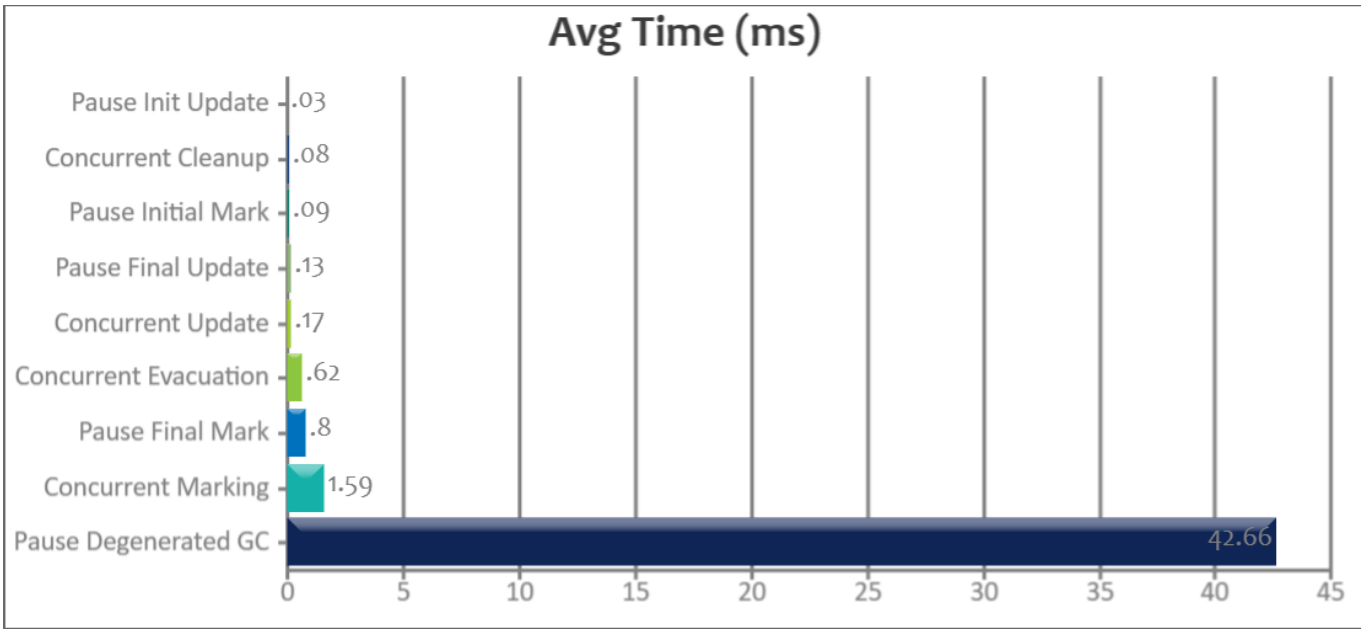
2 CPU Time : n/a

3 Latency:

Avg Pause GC Time	4.51 ms
Max Pause GC Time	364 ms

stw 时间平均值和最大值差异极大，吞吐量为 87 左右





到后期全部是退化 GC，并行阶段大幅减少