GC Intelligence Report

关注 872

gc.log

Duration: 32 sec 207 ms

Download

()

Share Report

Learn key sections

(https://www.youtube.com/watch?v=dN7S1RoKNYo)

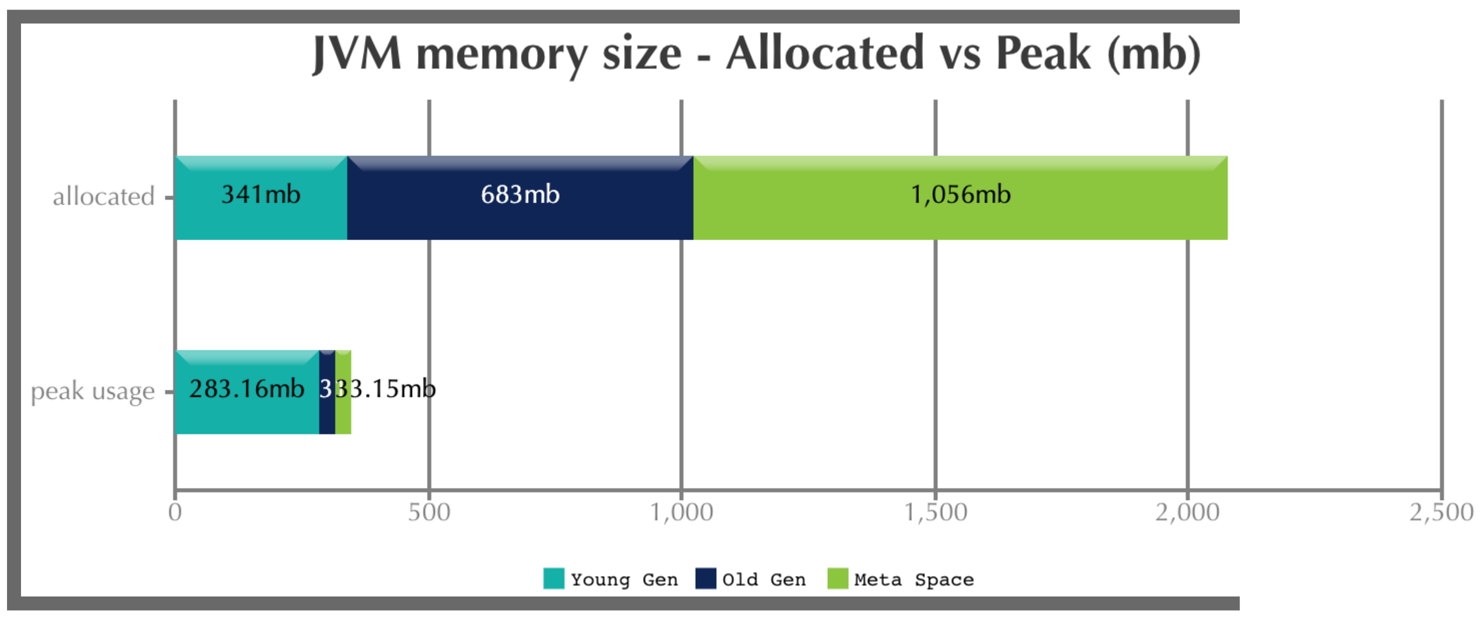
Recommendations

Our Machine Learning algorithms have identified memory optimization recommendations. To see those recommendations become our paid subscriber.

Select Plan (gc-subscription.jsp)

JVM memory size

Generation	Allocated	Peak
Young Generation	341 mb	283.16 mb
Old Generation	683 mb	31.08 mb
Meta Space	1.03 gb	33.15 mb
Young + Old + Meta space	2.03 gb	342.52 mb



Key Performance Indicators

(Important section of the report. To learn more about KPIs, click here (https://blog.gceasy.io/2016/10/01/garbage-collection-kpi/))

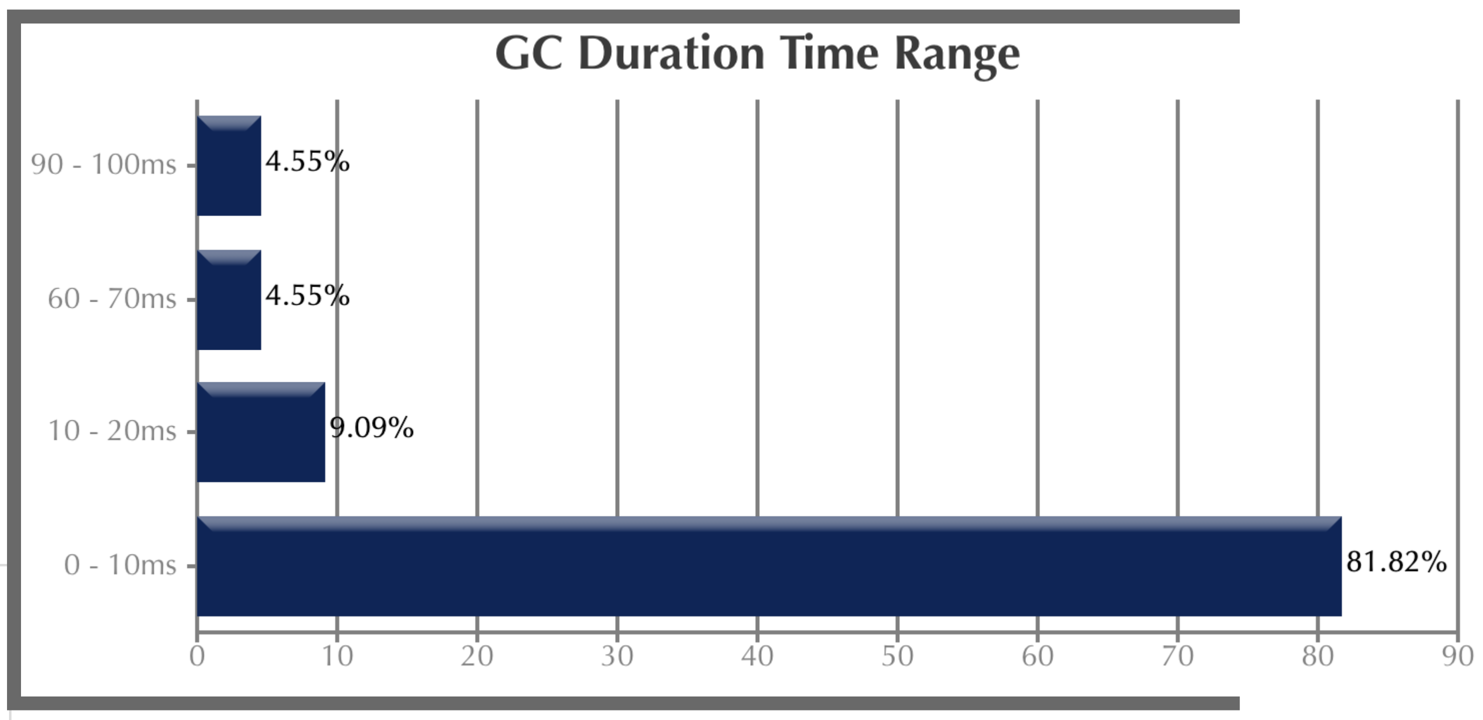
1 **Throughput : 99.255%**

2 **Latency:**

Avg Pause GC Time	**10.9 ms**
Max Pause GC Time	**90.0 ms**

GC **Pause** Duration Time Range :

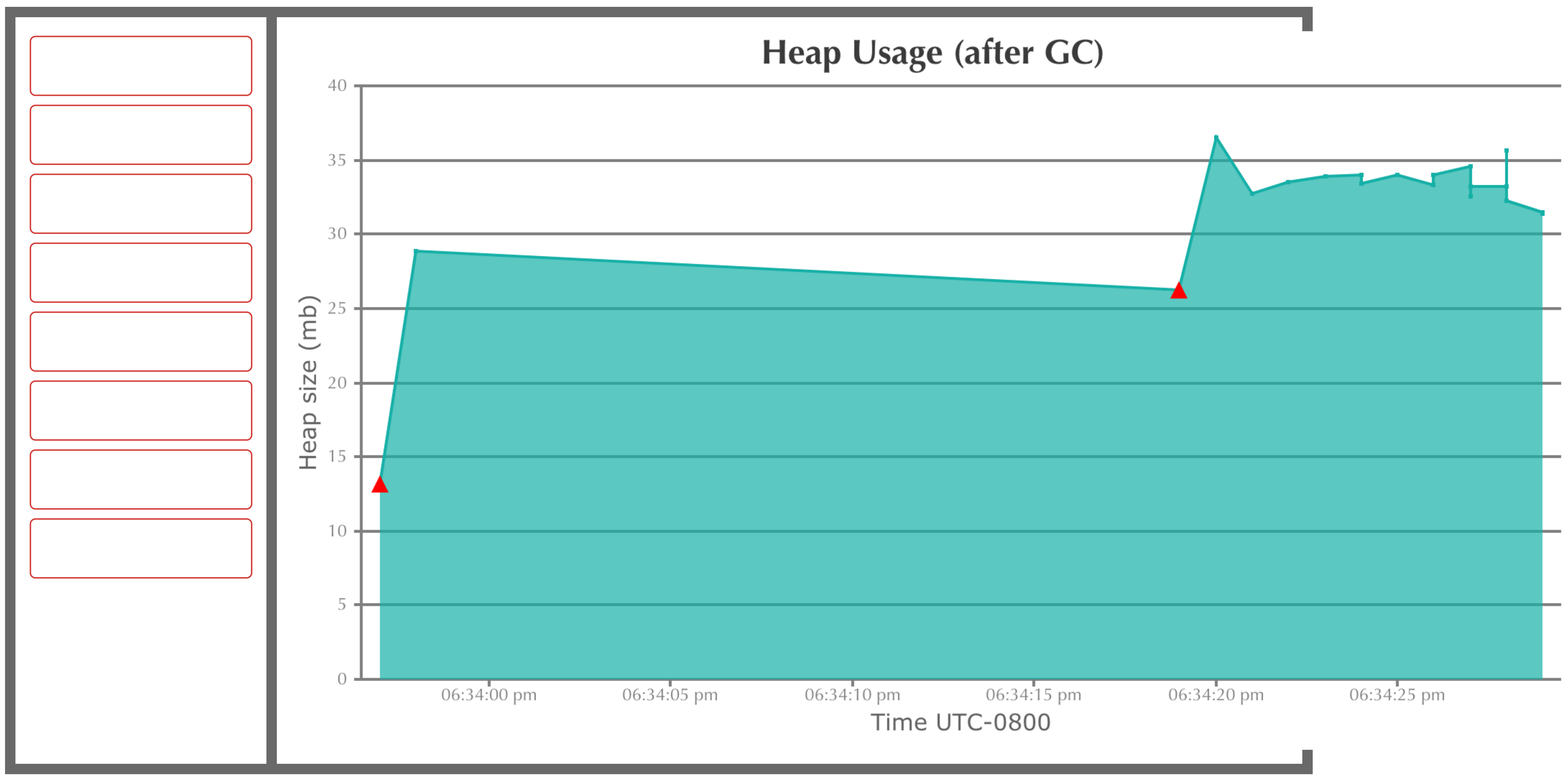
Duration (ms) 10 ms	No. of GCs	Percentage
0 - 10	18	81.82%
10 - 20	2	9.09%
60 - 70	1	4.55%
90 - 100	1	4.55%



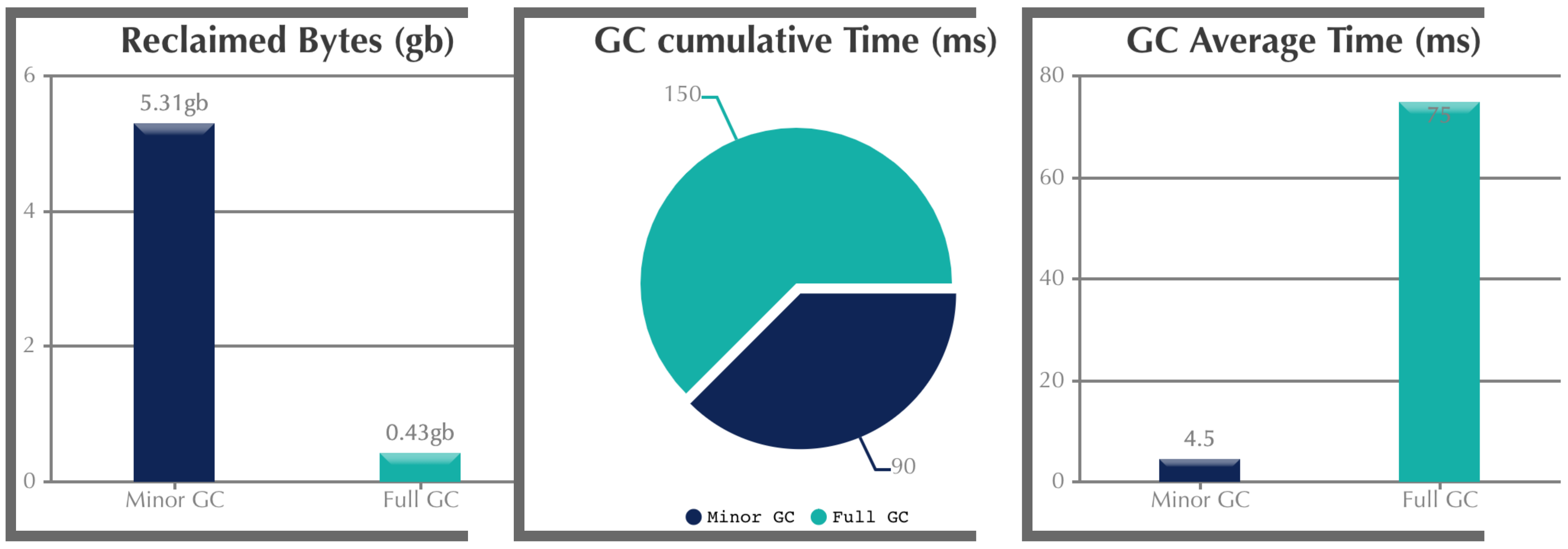
Interactive Graphs

(How to zoom graphs?) (https://www.youtube.com/watch?v=JhZFj6gJQyk)

(What is this graph?)



GC Statistics



Total GC stats

Total GC count	22
Total reclaimed bytes	5.74 gb
Total GC time	240 ms
Avg GC time	10.9 ms
GC avg time std dev	21.7 ms
GC min/max time	0 / 90.0 ms
GC Interval avg time	1 sec 533 ms

Minor GC stats

Minor GC count	20
Minor GC reclaimed	5.31 gb
Minor GC total time	90.0 ms
Minor GC avg time	4.50 ms
Minor GC avg time std dev	6.69 ms
Minor GC min/max time	0 / 20.0 ms
Minor GC Interval avg	1 sec 628 ms

Full GC stats

Full GC Count	2
Full GC reclaimed	444.85 mb
Full GC total time	150 ms
Full GC avg time	75.0 ms
Full GC avg time std dev	15.0 ms
Full GC min/max time	60.0 ms / 90.0 ms
Full GC Interval avg	22 sec 367 ms

GC Pause Statistics

Pause Count	22
Pause total time	240 ms
Pause avg time	10.9 ms
Pause avg time std dev	0.0
Pause min/max time	0 / 90.0 ms

Object Stats

(These are perfect micro-metrics (https://blog.gceasy.io/2017/05/30/improving-your-performance-reports/) to include in your performance reports)

Total created bytes	5.77 gb
Total promoted bytes	4.93 mb
Avg creation rate	183.59 mb/sec
Avg promotion rate	156 kb/sec

Memory Leak

No major memory leaks.

(**Note:** there are 8 flavours of OutOfMemoryErrors (https://tier1app.files.wordpress.com/2014/12/outofmemoryerror2.pdf). With GC Logs you can diagnose only 5 flavours of them(Java heap space, GC overhead limit exceeded, Requested array size exceeds VM limit, Permgen space, Metaspace). So in other words, your application could be still suffering from memory leaks, but need other tools to diagnose them, not just GC Logs.)

Consecutive Full GC

None.

Long Pause

None.

Safe Point Duration

(To learn more about SafePoint duration, click here (./gc-recommendations/safe-point-solution.jsp))

Not Reported in the log.

Allocation stall metrics

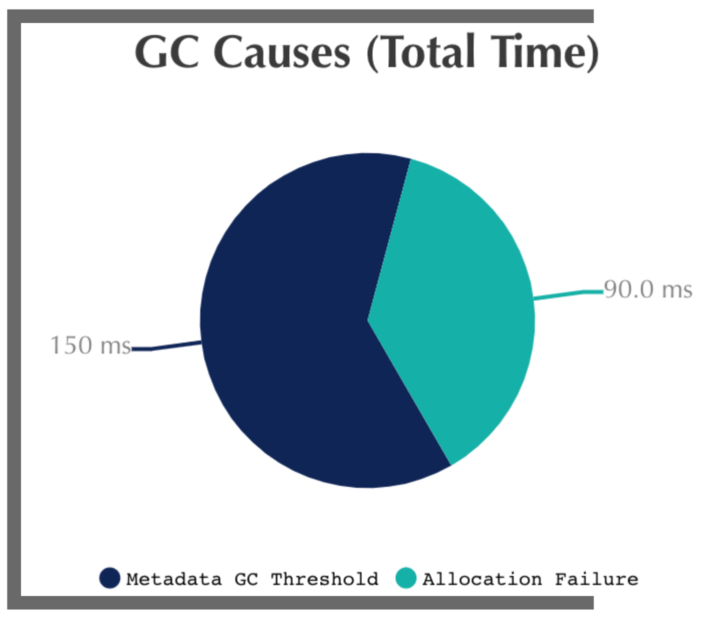
(To learn more about Allocation Stall, click here (./gc-recommendations/allocation-stall-solution.jsp))

Not Reported in the log.

GC Causes

(What events caused GCs & how much time they consumed?)

Cause	Count	Avg Time	Max Time	Total Time
Metadata GC Threshold	2	75.0 ms	90.0 ms	150 ms
Allocation Failure	20	4.50 ms	20.0 ms	90.0 ms



Tenuring Summary

Not reported in the log.

Command Line Flags

-XX:InitialHeapSize=1073741824 -XX:MaxHeapSize=1073741824 -XX:MaxNewSize=357564416 -XX:NewSize=357564416 -XX:+PrintGC -XX:+PrintGCDateStamps -XX:+PrintGCDetails -XX:+PrintGCTimeStamps -XX:ThreadStackSize=1024 -XX:+UseCompressedClassPointers -XX:+UseCompressedOops -XX:+UseParNewGC

Become a DevOps champion in your organization

(Best practises/tools)

- ✓ Use **fastThread.io** (https://fastthread.io/) tool to analyze thread dumps, core dumps and hs_err_pid files
- ✓ Use **HeapHero.io** (https://heaphero.io/) tool to analyze heap dumps
- ✓ Do proactive Garbage Collection analysis on all your JVMs (not just one or two) using the GC log analysis API (https://blog.gceasy.io/2016/06/18/garbage-collection-log-analysis-api/)
- ✓ Purchase 'Enterprise' edition (http://gceasy.io/pricing.jsp) for 10x fast, unlimited, secure usage

Do you like this report?

GCeasy

GCeasy is the industry's first online Garbage collection log analysis tool aided by Machine Learning.

It's used by thousands of enterprises globally to tune & troubleshoot complex memory & GC problems.

Reach Us

Dublin, CA, USA

+ 1-415-948-5431

team@tier1app.com (mailto:team@tier1app.com)

Quick Links

- Terms & Conditions (terms.jsp)
- Privacy policy (gc-privacy.jsp)
- **fastThread (sister product) (https://fastthread.io/)**
- **HeapHero (sister product) (https://heaphero.io/)**
- **yCrash (sister product) (https://ycrash.io/)**

Stay in Touch!

Follow us on our social networks!

(https://www.facebook.com/tier1app)

(https://twitter.com/tier1app)

(https://www.linkedin.com/company/gceasy)

(https://www.youtube.com/channel/UCM-yObJ7pBjEy1wJMq5bDdw)

Made by Tier1app (http://tier1app.com) with + soul + intelligence