DEVELOPER LIVE

ZGC: The Future of Low-Latency Garbage Collection Is Here

Per Liden

Consulting Member of Technical Staff, Oracle

Safe Harbor Statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, timing, and pricing of any features or functionality described for Oracle's products may change and remains at the sole discretion of Oracle Corporation.

Copyright © 2022, Oracle and/or its affiliates

A Scalable Low-Latency Garbage Collector

Copyright © 2022, Oracle and/or its affiliates

Properties

Max GC pause time

1ms

Multi-terabyte heaps

TB

Copyright © 2022, Oracle and/or its affiliates

What’s the Catch?

Expect some reduction in throughput

Copyright © 2022, Oracle and/or its affiliates

Goal

Easy to tune!

Copyright © 2022, Oracle and/or its affiliates

GC Landscape

Oracle supported garbage collectors

GC	Optimized For
Serial	Memory Footprint
Parallel	Throughput
G1	Throughput/Latency Balance
ZGC	Low Latency

Copyright © 2022, Oracle and/or its affiliates

ZGC at a Glance

Concurrent
Tracing
Compacting
Single generation

Region-based
NUMA-aware
Load barriers
Colored pointers

Copyright © 2022, Oracle and/or its affiliates

ZGC pauses are O(1)

Copyright © 2022, Oracle and/or its affiliates

Available on All Commonly Used Platforms

Linux

x86 / Arm / PPC

Windows

x86 / Arm

macOS

x86 / Arm

(64-bit)

Copyright © 2022, Oracle and/or its affiliates

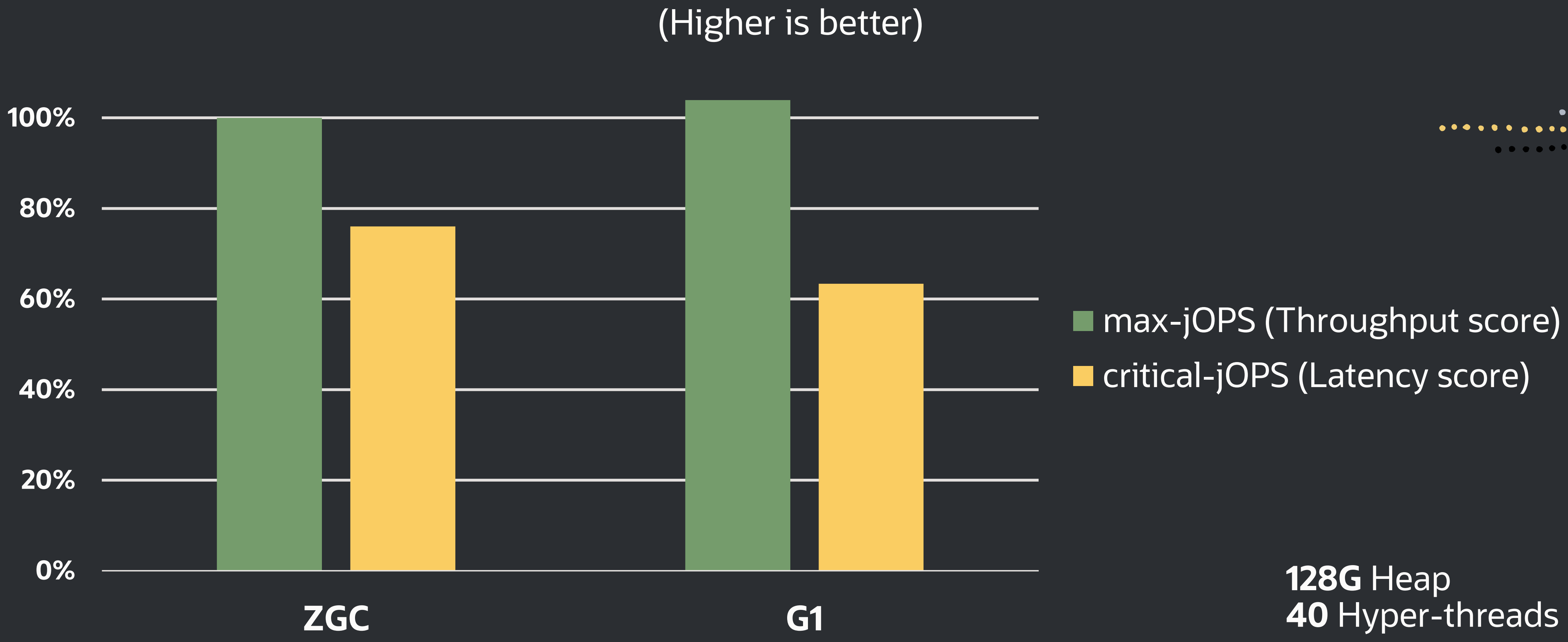
Production Ready since JDK 15

Copyright © 2022, Oracle and/or its affiliates

Performance

Copyright © 2022, Oracle and/or its affiliates

SPECjbb2015 – Benchmark Score

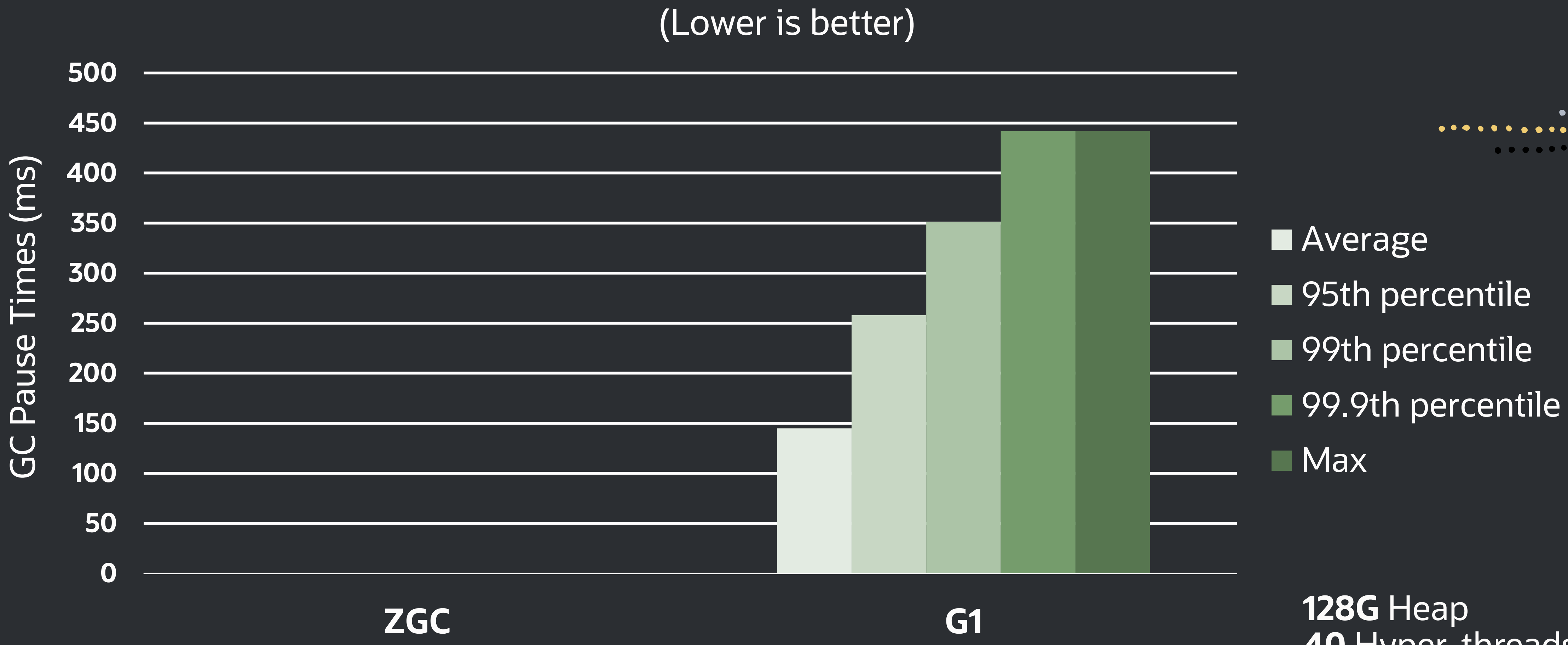


128G Heap
40 Hyper-threads (Intel)

Copyright © 2022, Oracle and/or its affiliates

SPECjbb®2015 is a registered trademark of the Standard Performance Evaluation Corporation (spec.org). The actual results are not represented as compliant because the SUT may not meet SPEC's requirements for general availability.

SPECjbb2015 – GC Pause Times

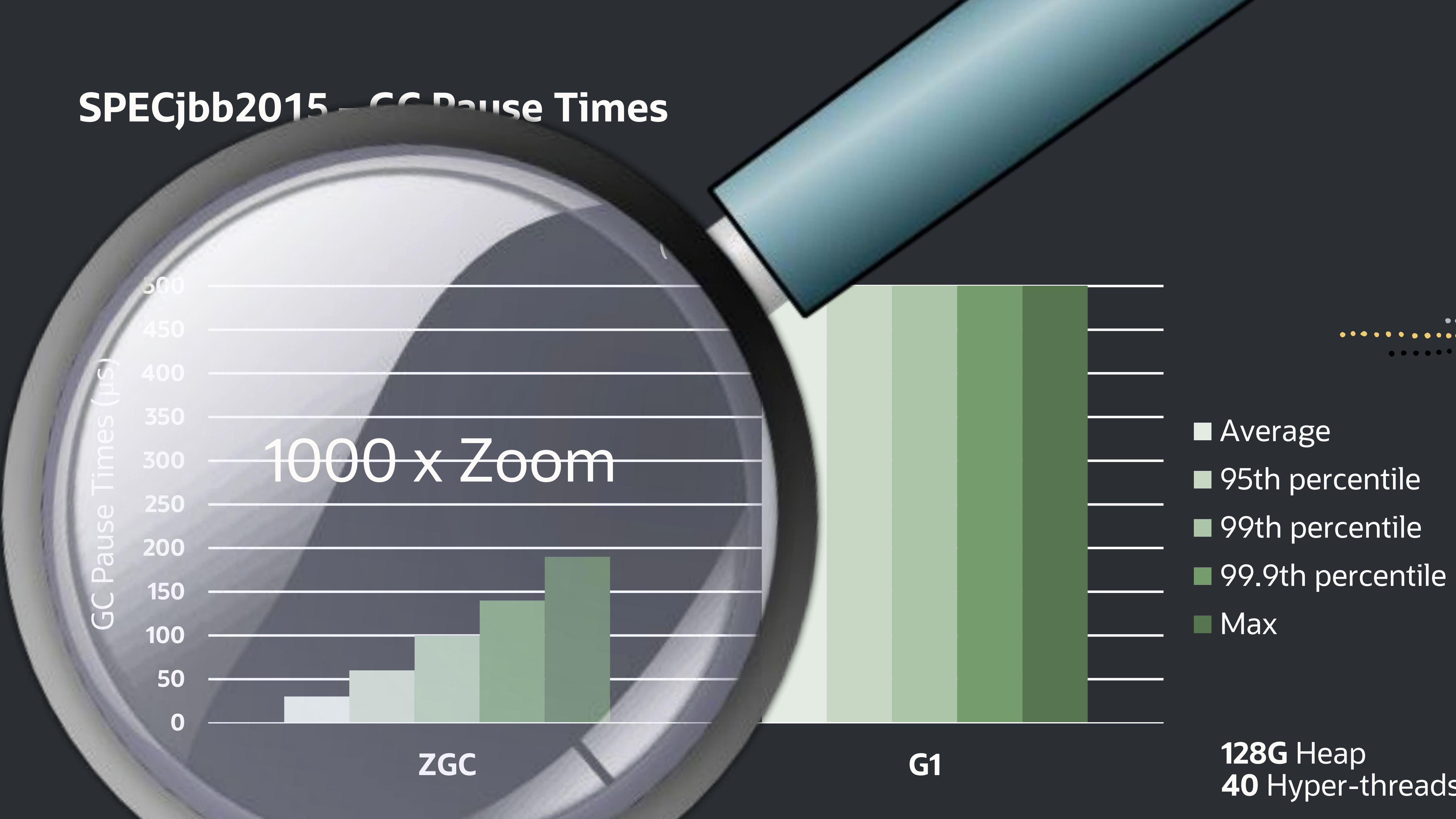


128G Heap
40 Hyper-threads (Intel)

Copyright © 2022, Oracle and/or its affiliates

SPECjbb®2015 is a registered trademark of the Standard Performance Evaluation Corporation (spec.org). The actual results are not represented as compliant because the SUT may not meet SPEC's requirements for general availability.

SPECjbb2015 – GC Pause Times

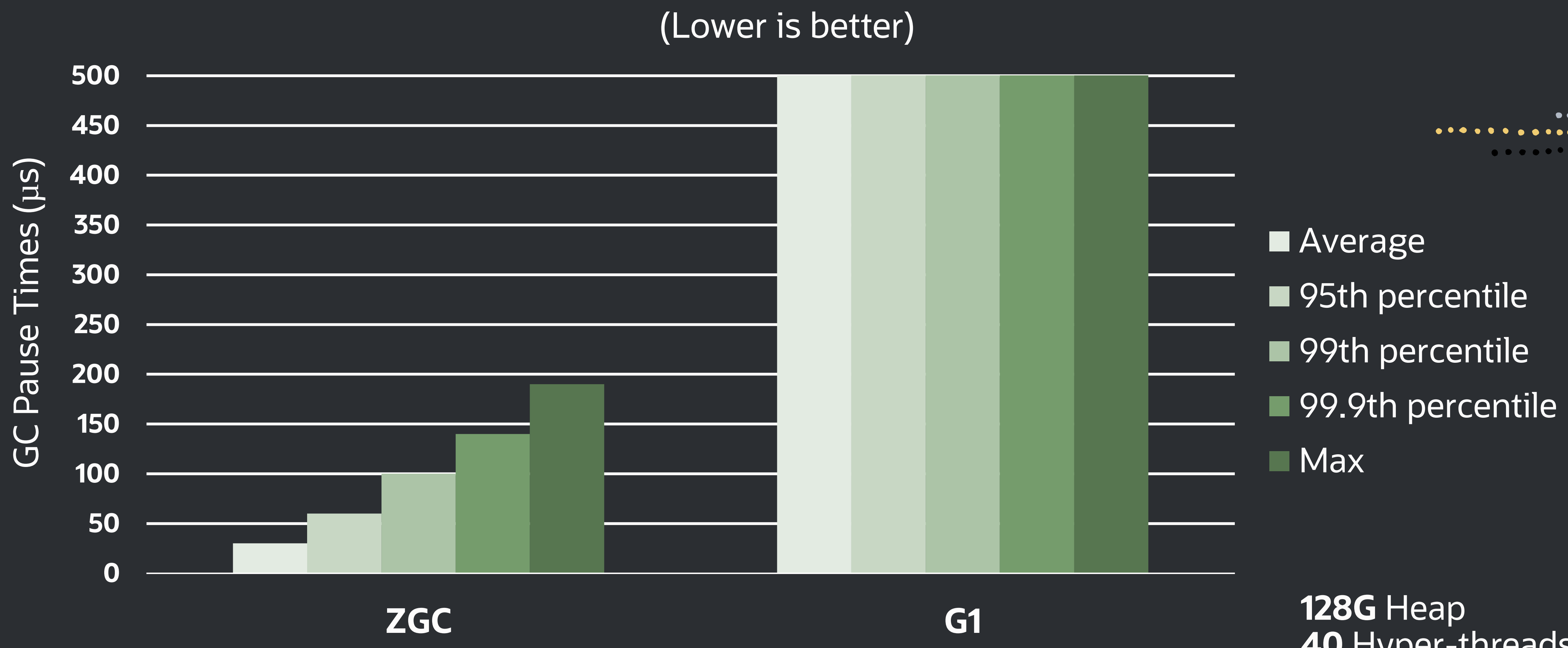


128G Heap
40 Hyper-threads (Intel)

SPECjbb®2015 is a registered trademark of the Standard Performance Evaluation Corporation (spec.org). The actual results are not represented as compliant because the SUT may not meet SPEC's requirements for general availability.

Copyright le and/or its affiliates

SPECjbb2015 – GC Pause Times



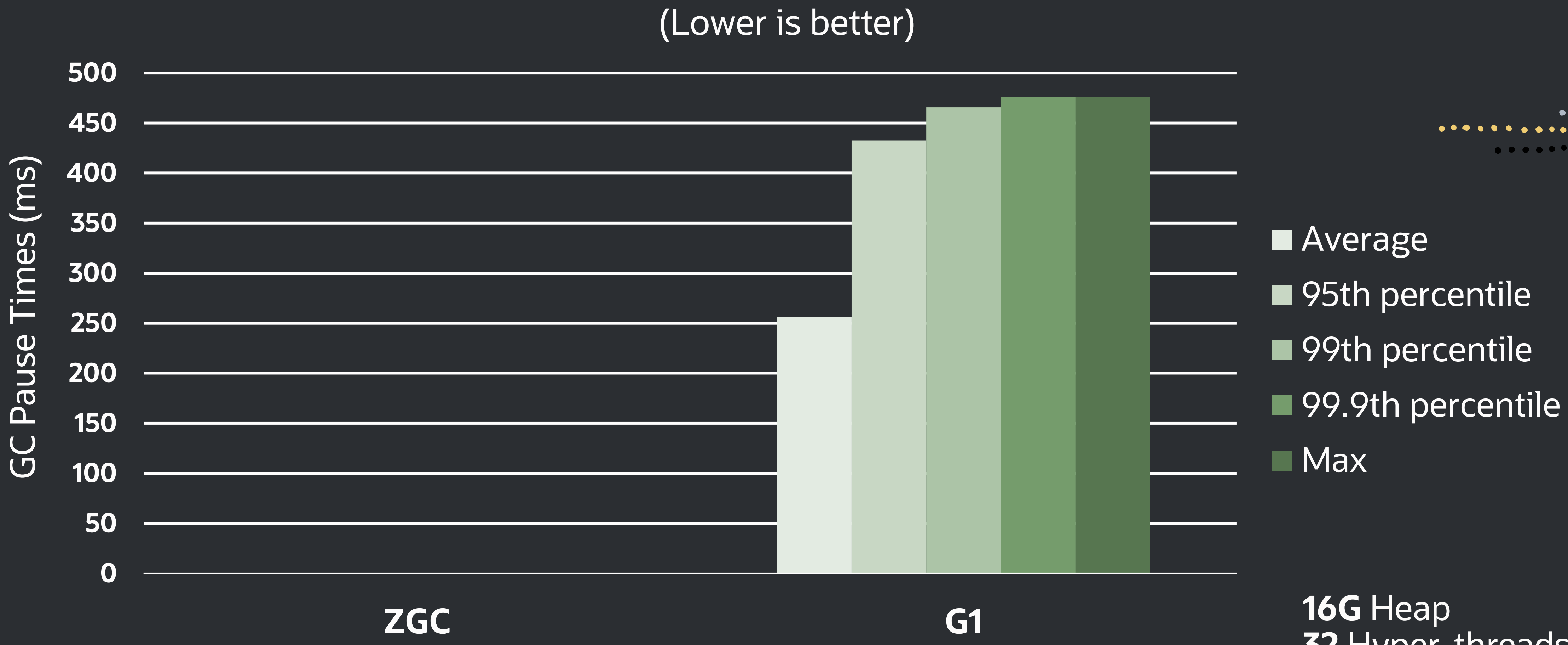
128G Heap
40 Hyper-threads (Intel)

Copyright © 2022, Oracle and/or its affiliates

SPECjbb®2015 is a registered trademark of the Standard Performance Evaluation Corporation (spec.org). The actual results are not represented as compliant because the SUT may not meet SPEC's requirements for general availability.

BigRamTester – GC Pause Times

Lots of heap fragmentation

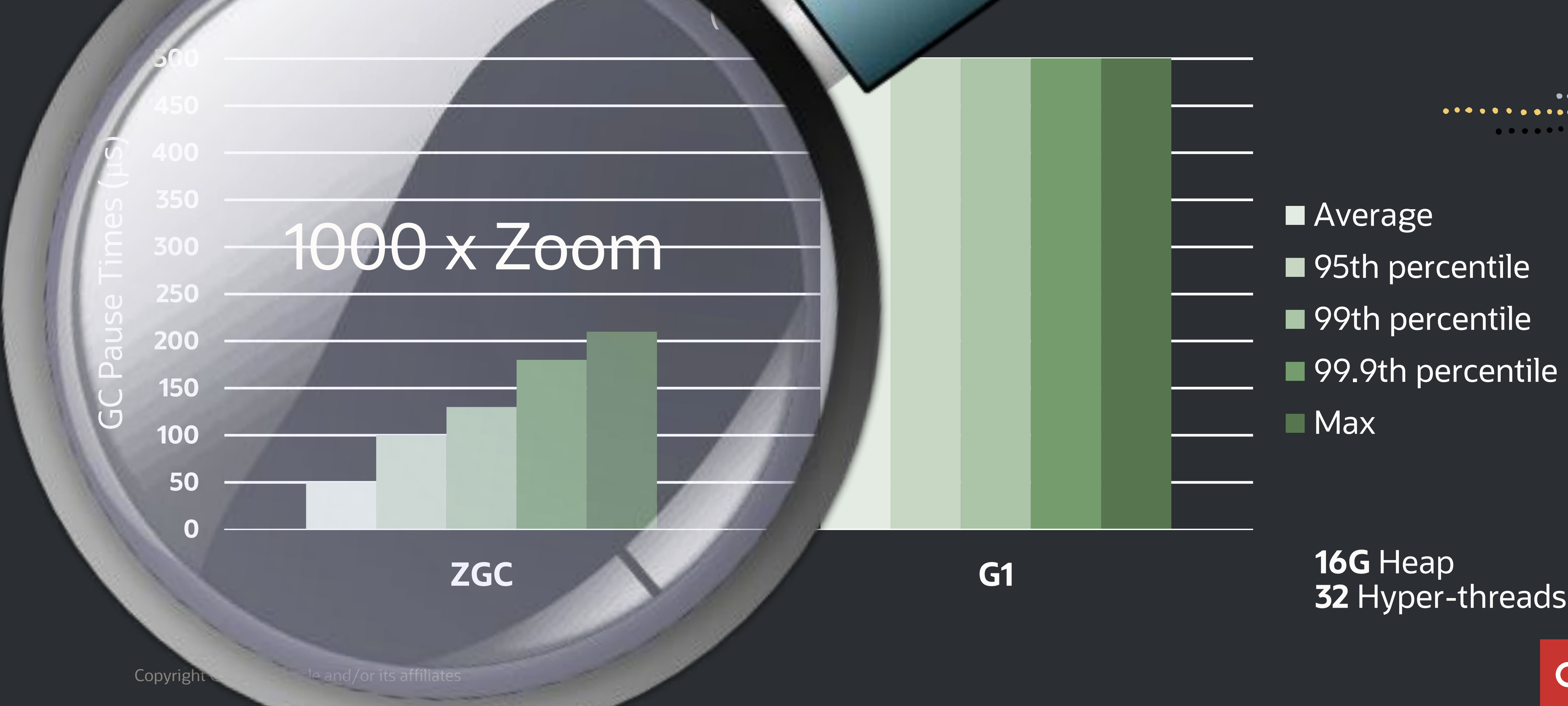


16G Heap
32 Hyper-threads (Intel)

Copyright © 2022, Oracle and/or its affiliates

BigRamTester – GC Pause Times

Lots of heap fragmentation

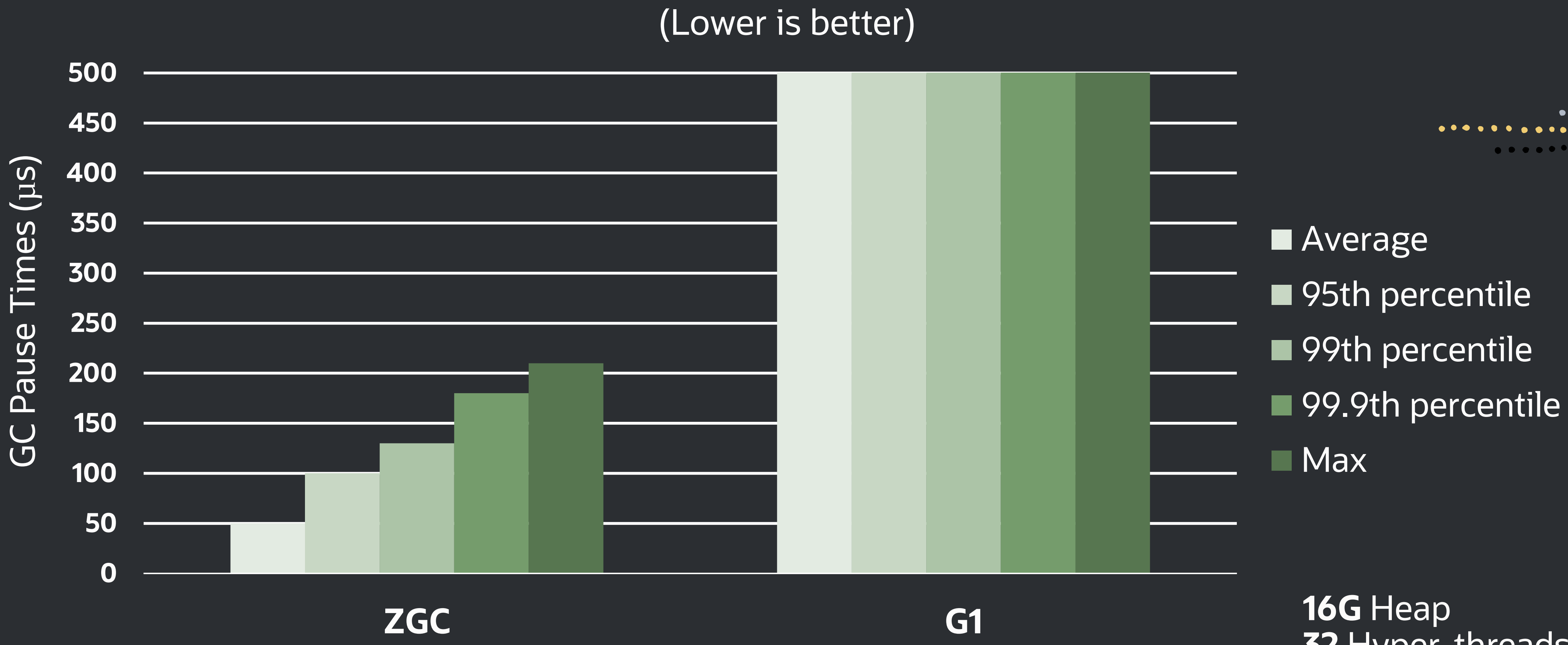


16G Heap
32 Hyper-threads (Intel)

Copyright © Oracle and/or its affiliates

BigRamTester – GC Pause Times

Lots of heap fragmentation

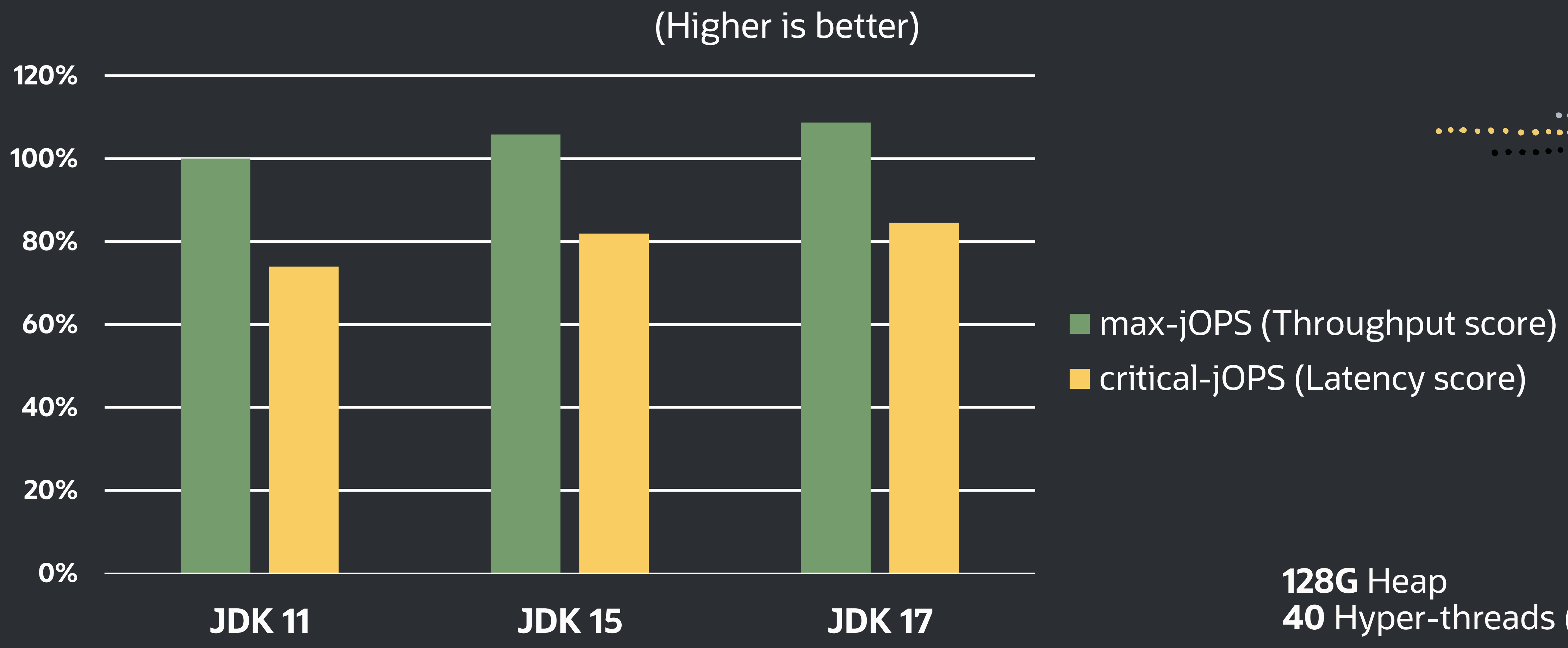


16G Heap
32 Hyper-threads (Intel)

Copyright © 2022, Oracle and/or its affiliates

ZGC Improvements Over Time

SPECjbb2015 – Benchmark Score



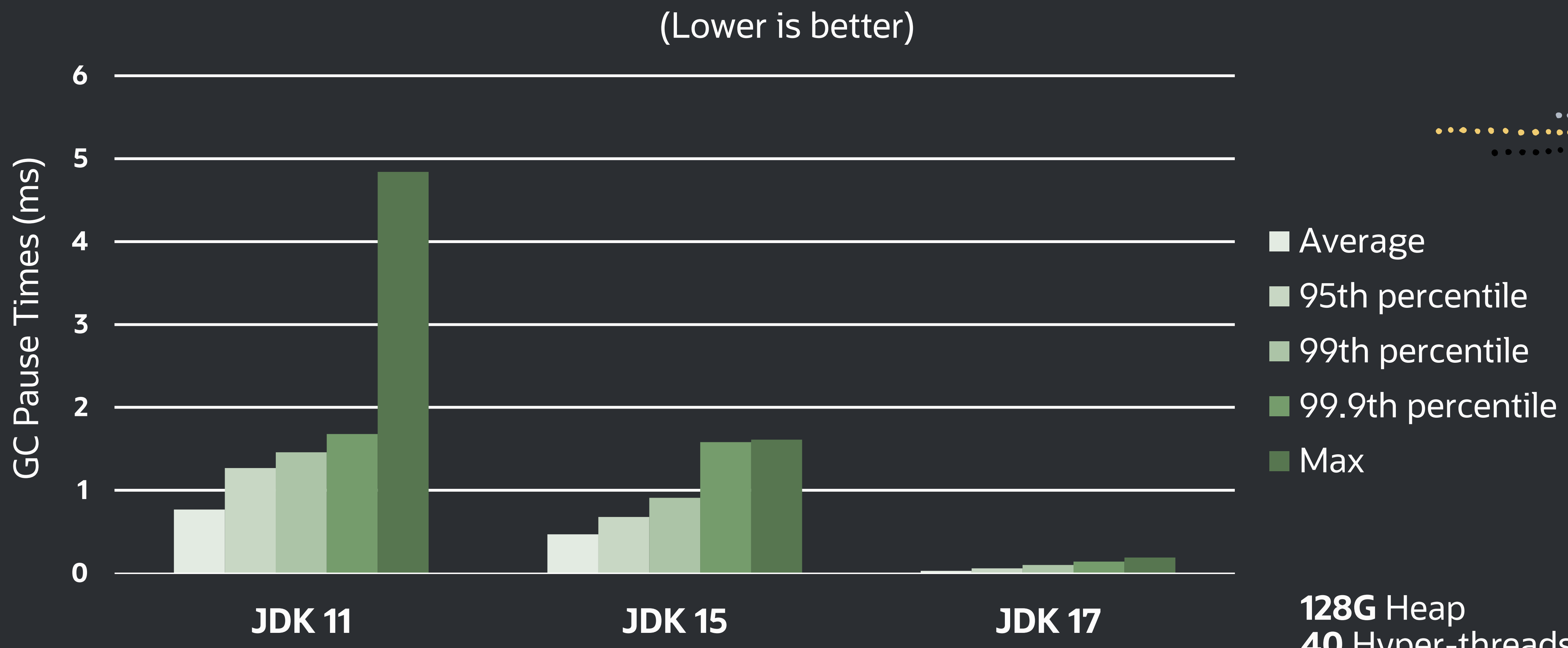
128G Heap
40 Hyper-threads (Intel)

Copyright © 2022, Oracle and/or its affiliates

SPECjbb®2015 is a registered trademark of the Standard Performance Evaluation Corporation (spec.org). The actual results are not represented as compliant because the SUT may not meet SPEC's requirements for general availability.

ZGC Improvements Over Time

SPECjbb2015 – GC Pause Times



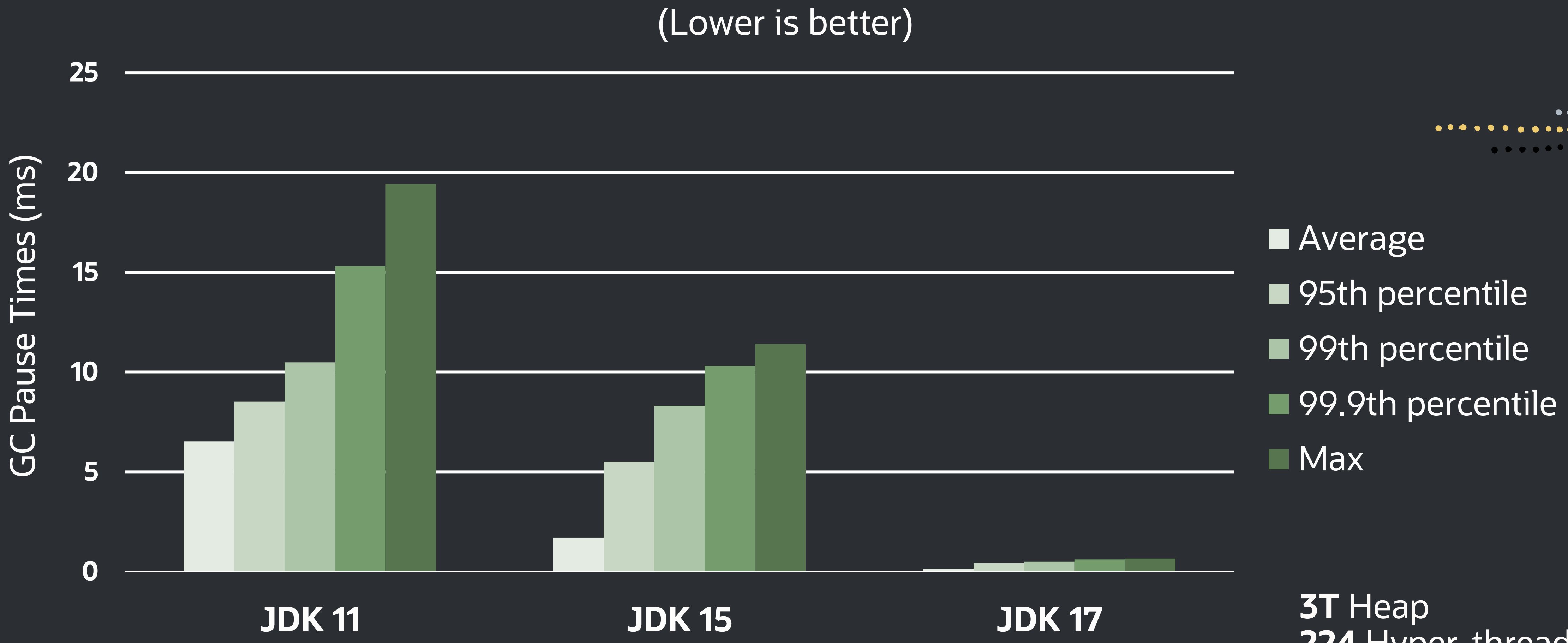
128G Heap
40 Hyper-threads (Intel)

Copyright © 2022, Oracle and/or its affiliates

SPECjbb®2015 is a registered trademark of the Standard Performance Evaluation Corporation (spec.org). The actual results are not represented as compliant because the SUT may not meet SPEC's requirements for general availability.

ZGC Improvements Over Time (Large System)

SPECjbb2015 – GC Pause Times



3T Heap
224 Hyper-threads (Intel)

Copyright © 2022, Oracle and/or its affiliates

SPECjbb®2015 is a registered trademark of the Standard Performance Evaluation Corporation (spec.org). The actual results are not represented as compliant because the SUT may not meet SPEC's requirements for general availability.

Using ZGC

Copyright © 2022, Oracle and/or its affiliates

Enable

```
-XX:+UseZGC
```

Copyright © 2022, Oracle and/or its affiliates

Tuning

Set Max Heap Size

```
-Xmx<size>
```

Copyright © 2022, Oracle and/or its affiliates

Logging

`-Xlog:gc` (basic)

`-Xlog:gc*` (detailed)

Copyright © 2022, Oracle and/or its affiliates

Roadmap

Copyright © 2022, Oracle and/or its affiliates

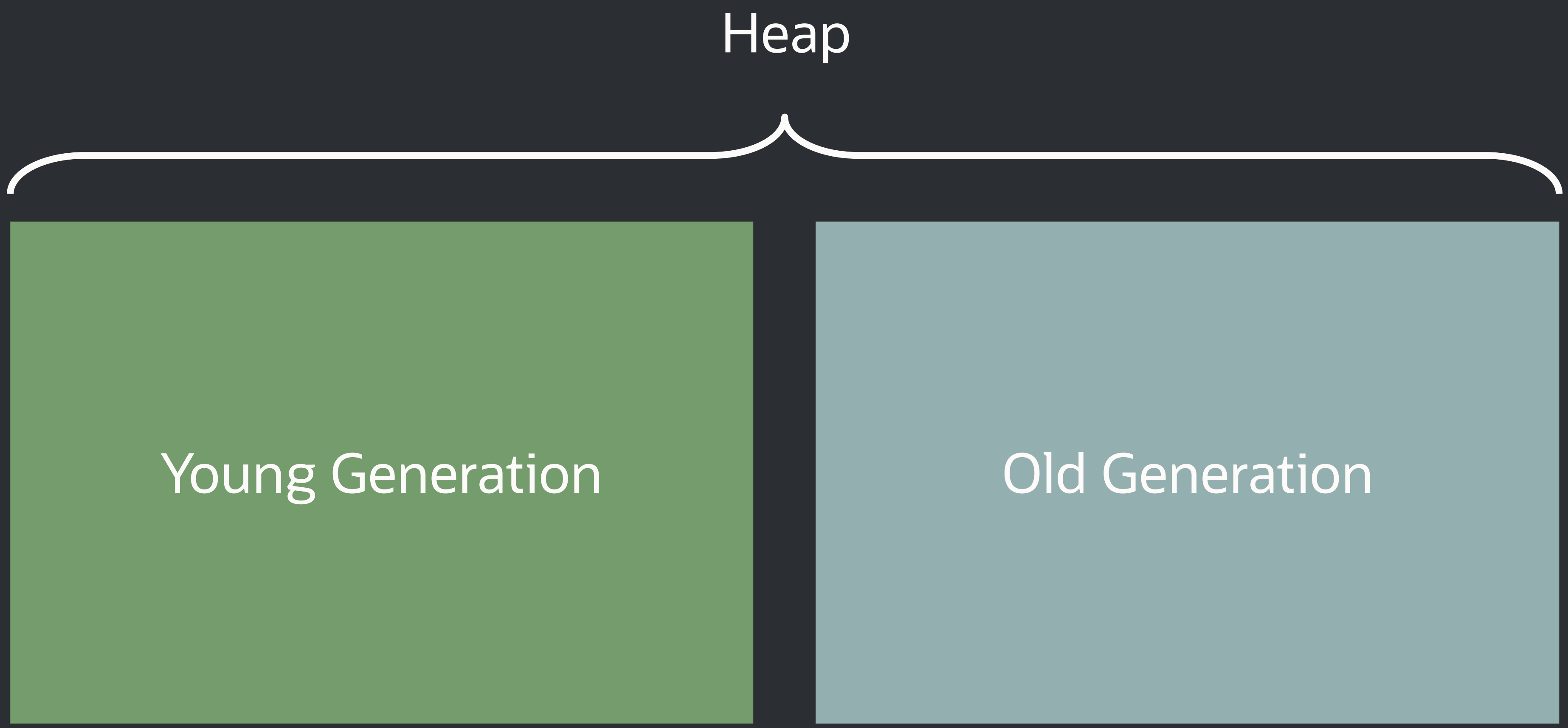
Generational ZGC

Copyright © 2022, Oracle and/or its affiliates

The Generational Hypothesis

A very common patten in Java applications is that ***most objects are short lived***

Copyright © 2022, Oracle and/or its affiliates

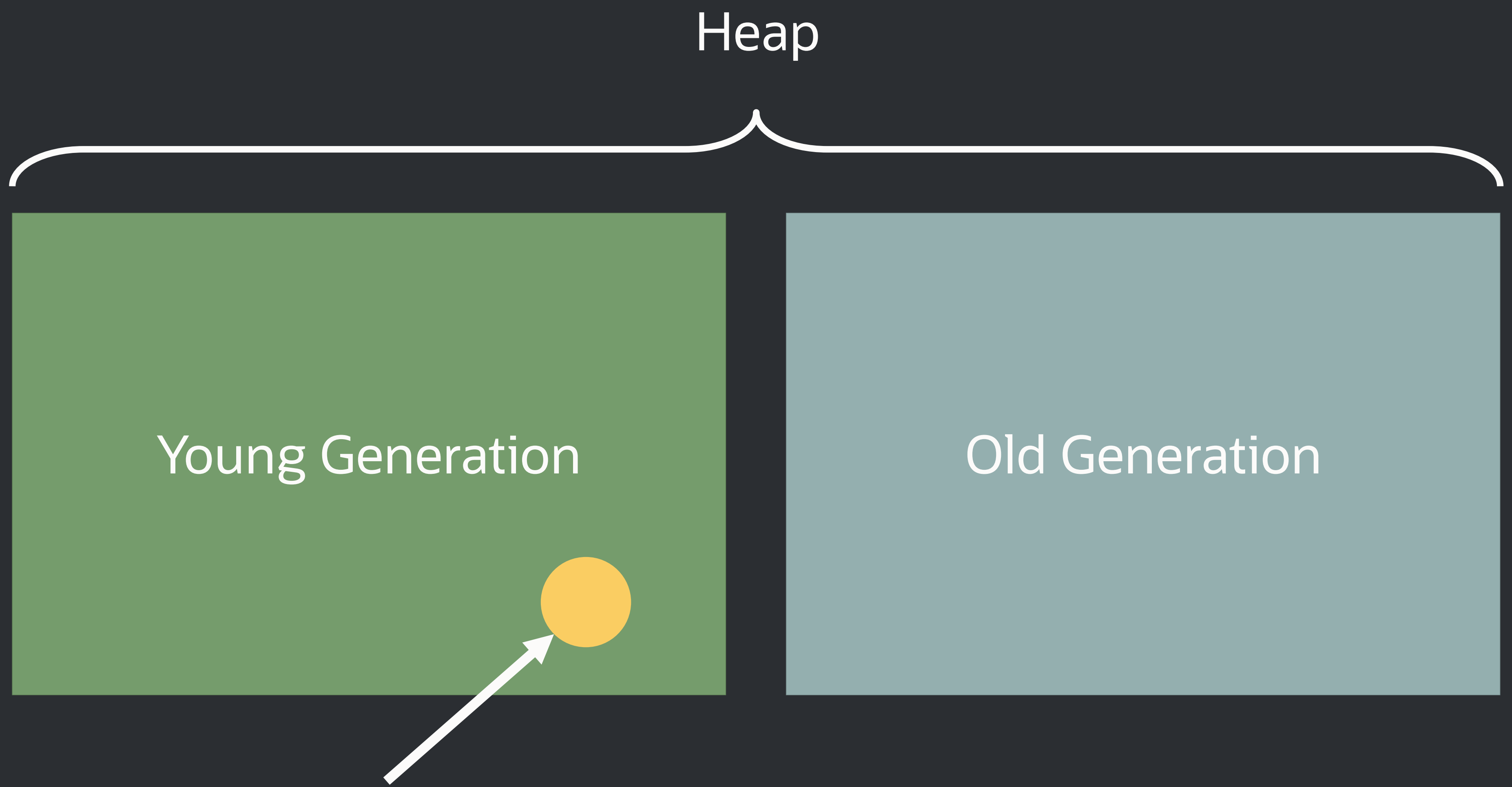


Heap

Young Generation

Old Generation

Copyright © 2022, Oracle and/or its affiliates



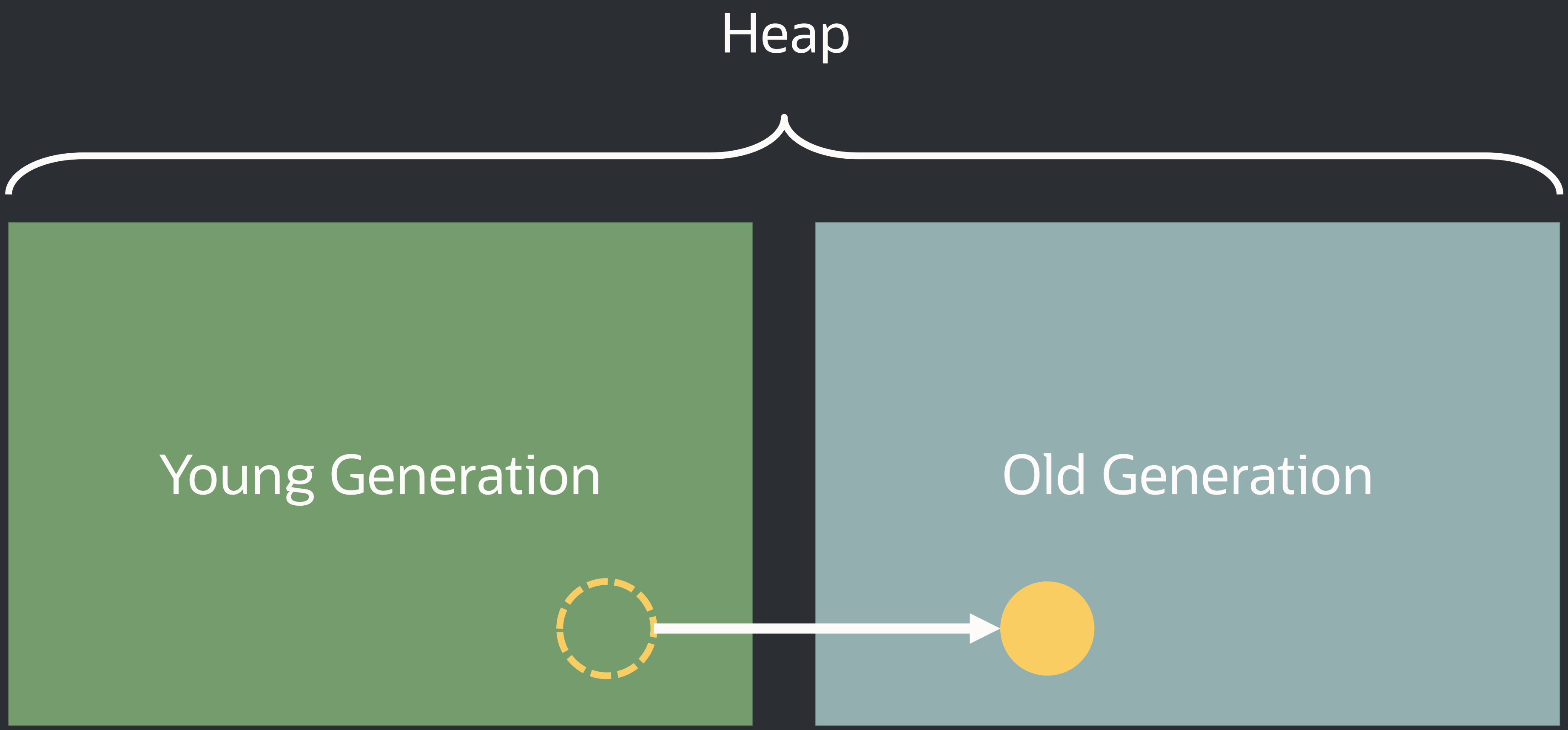
Heap

Young Generation

Old Generation

New object allocated in the young generation

Copyright © 2022, Oracle and/or its affiliates



Object promoted to the old generation

Copyright © 2022, Oracle and/or its affiliates

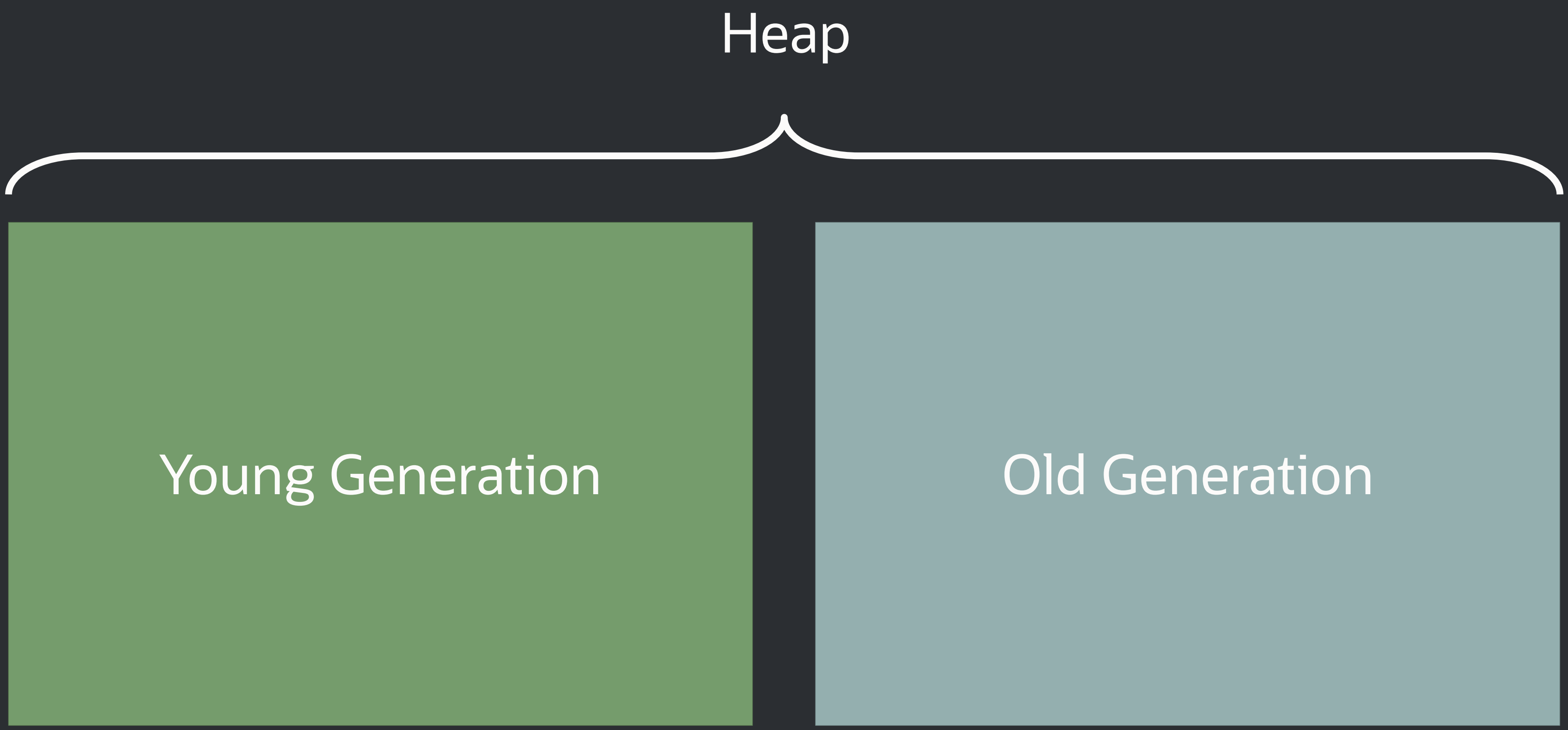
Reduced Effort to Collect Garbage

Withstand higher allocation rates

Lower heap headroom

Lower CPU usage

Copyright © 2022, Oracle and/or its affiliates



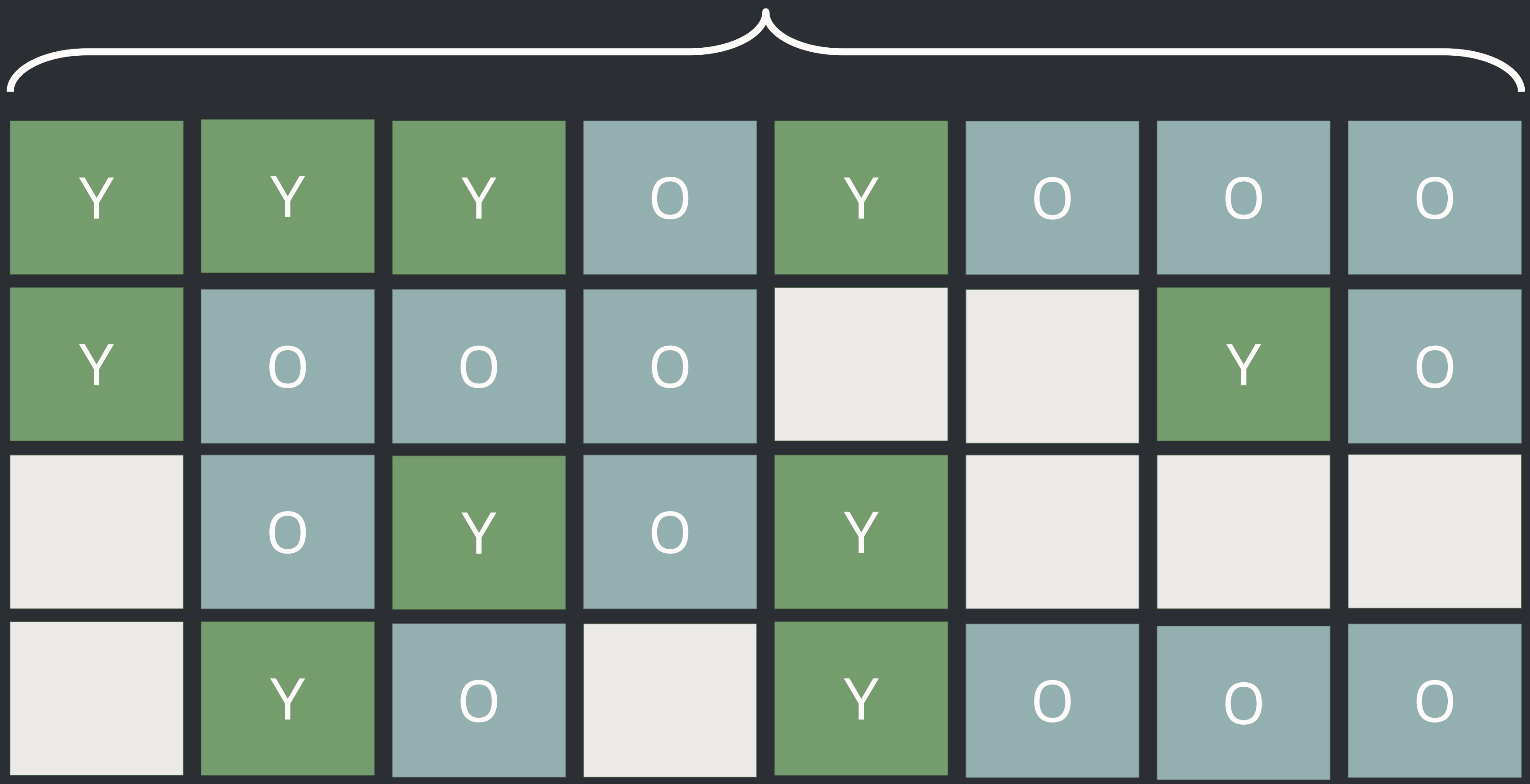
Heap

Young Generation

Old Generation

Copyright © 2022, Oracle and/or its affiliates

ZGC Heap

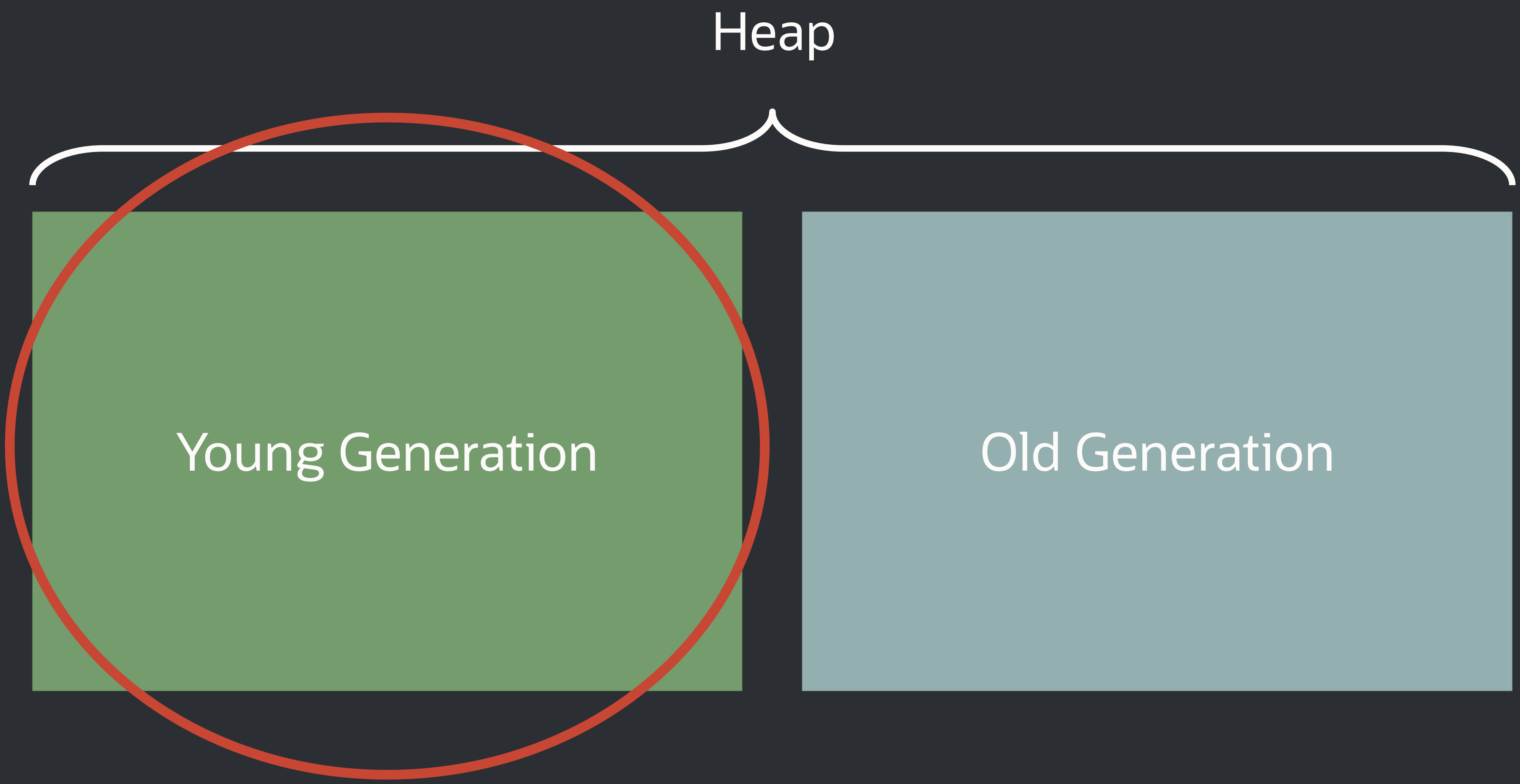


Y	Y	Y	O	Y	O	O	O
Y	O	O	O			Y	O
	O	Y	O	Y			
	Y	O		Y	O	O	O

Copyright © 2022, Oracle and/or its affiliates

Minor & Major Collections

Copyright © 2022, Oracle and/or its affiliates



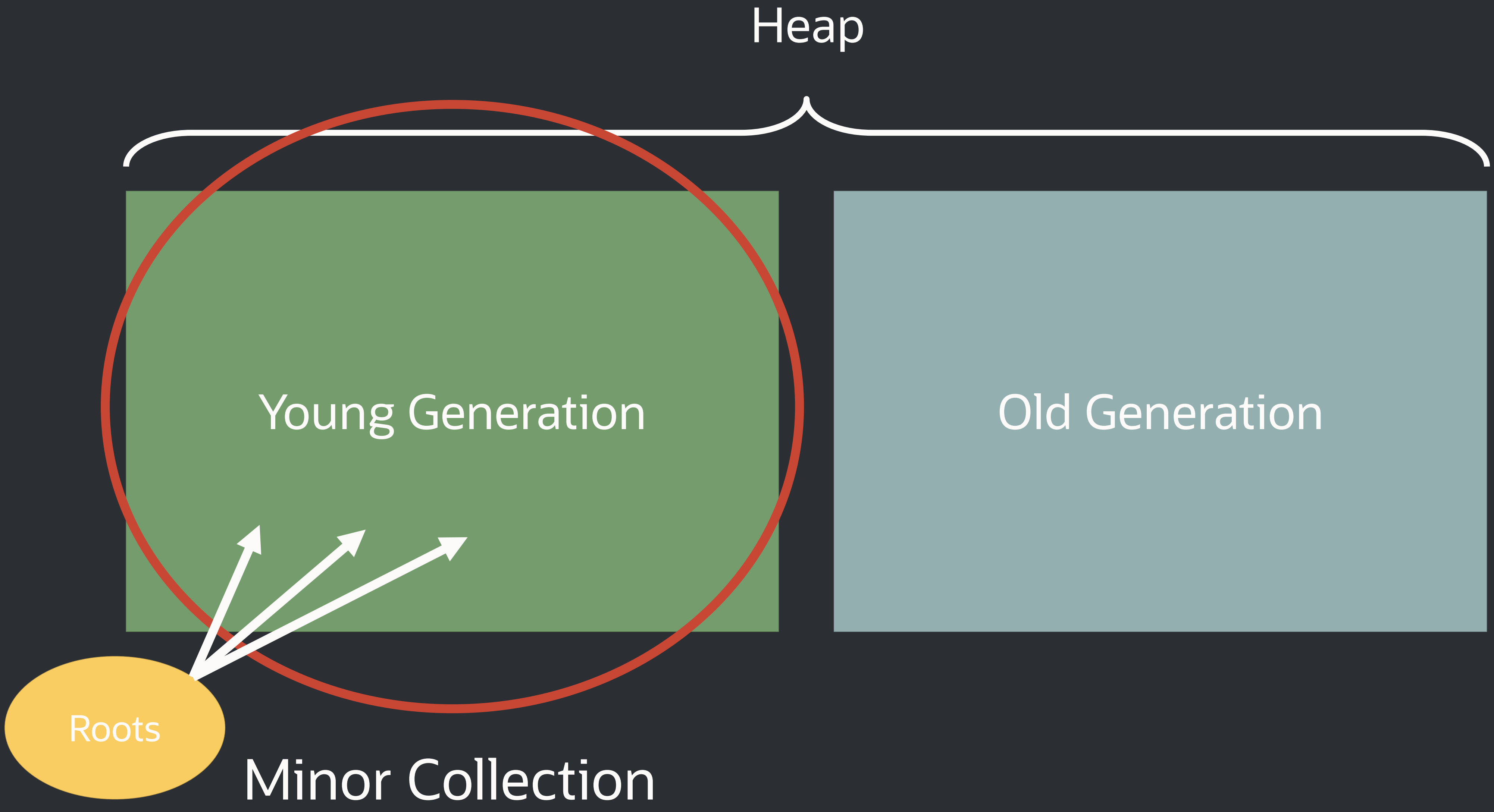
Heap

Young Generation

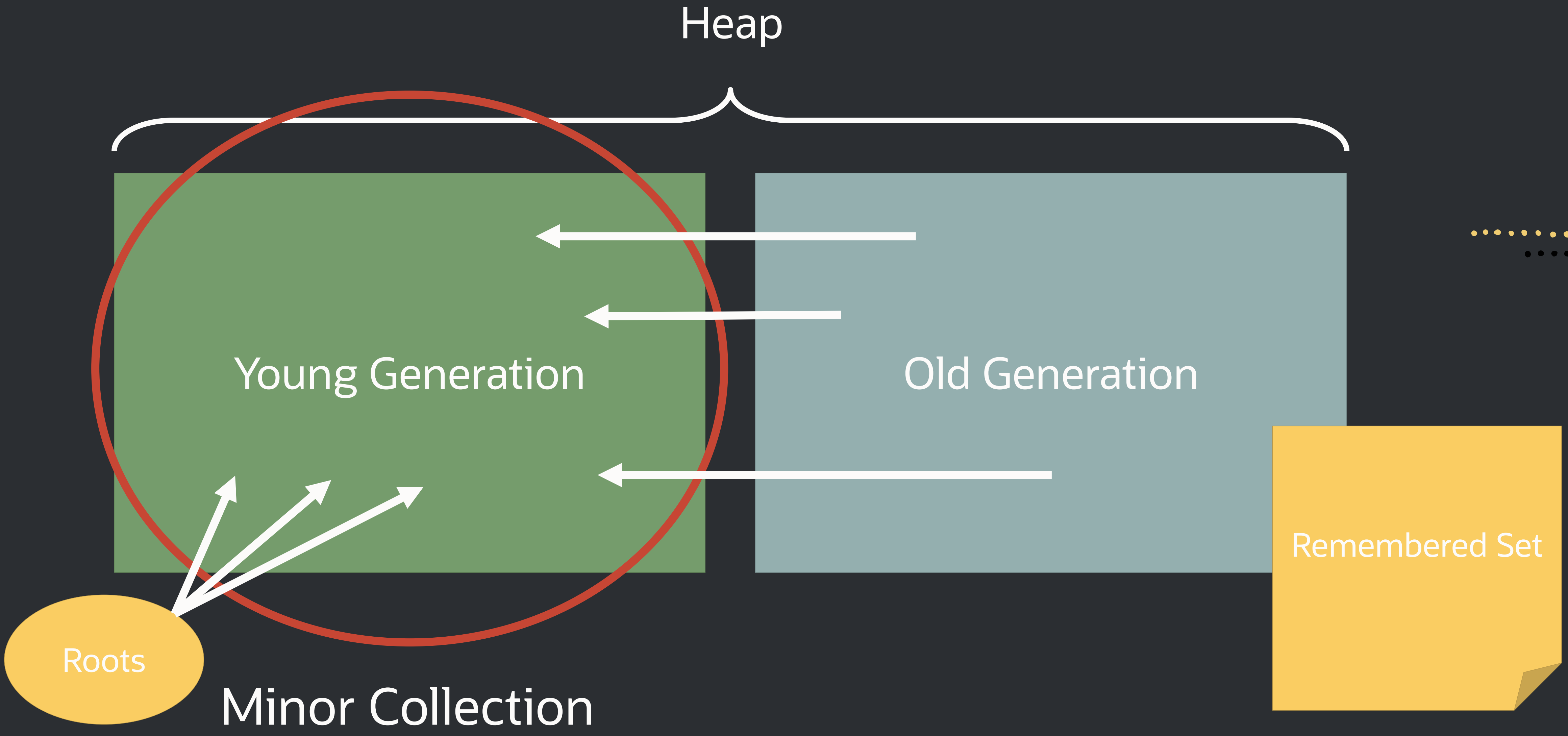
Old Generation

Minor Collection

Copyright © 2022, Oracle and/or its affiliates



Copyright © 2022, Oracle and/or its affiliates



Heap

Young Generation

Old Generation

Remembered Set

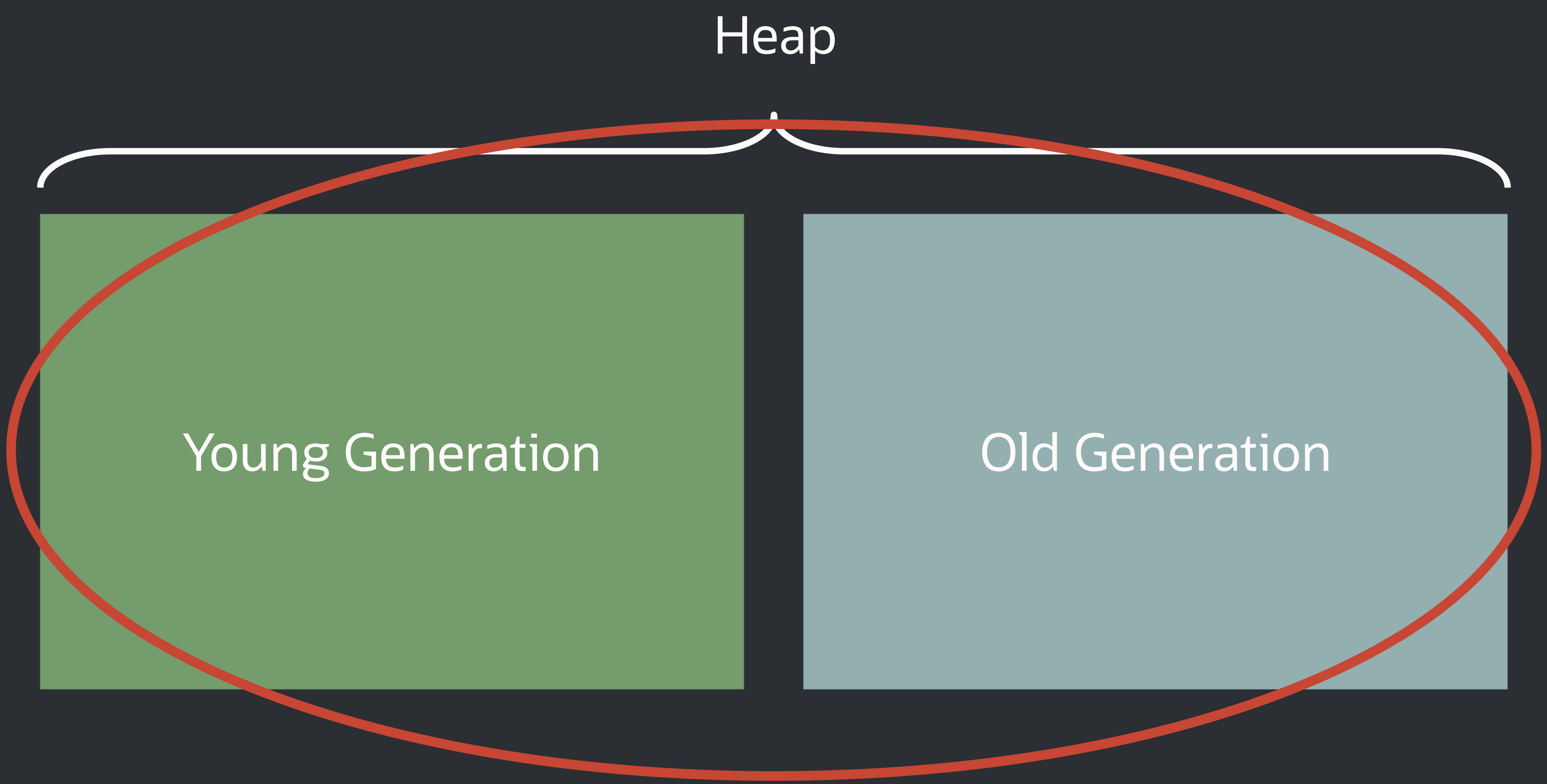
Roots

Minor Collection

Copyright © 2022, Oracle and/or its affiliates

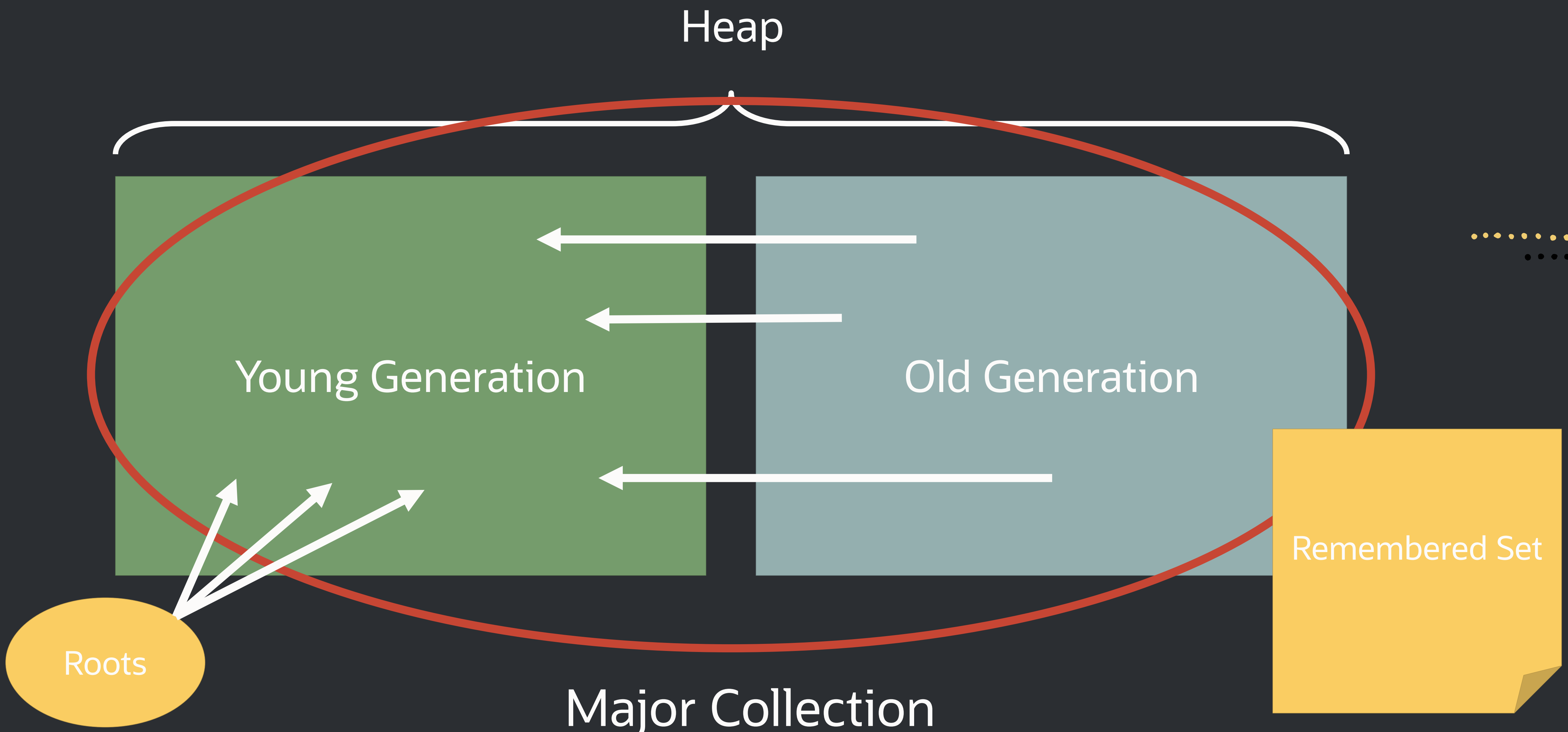
Major Collection

Copyright © 2022, Oracle and/or its affiliates



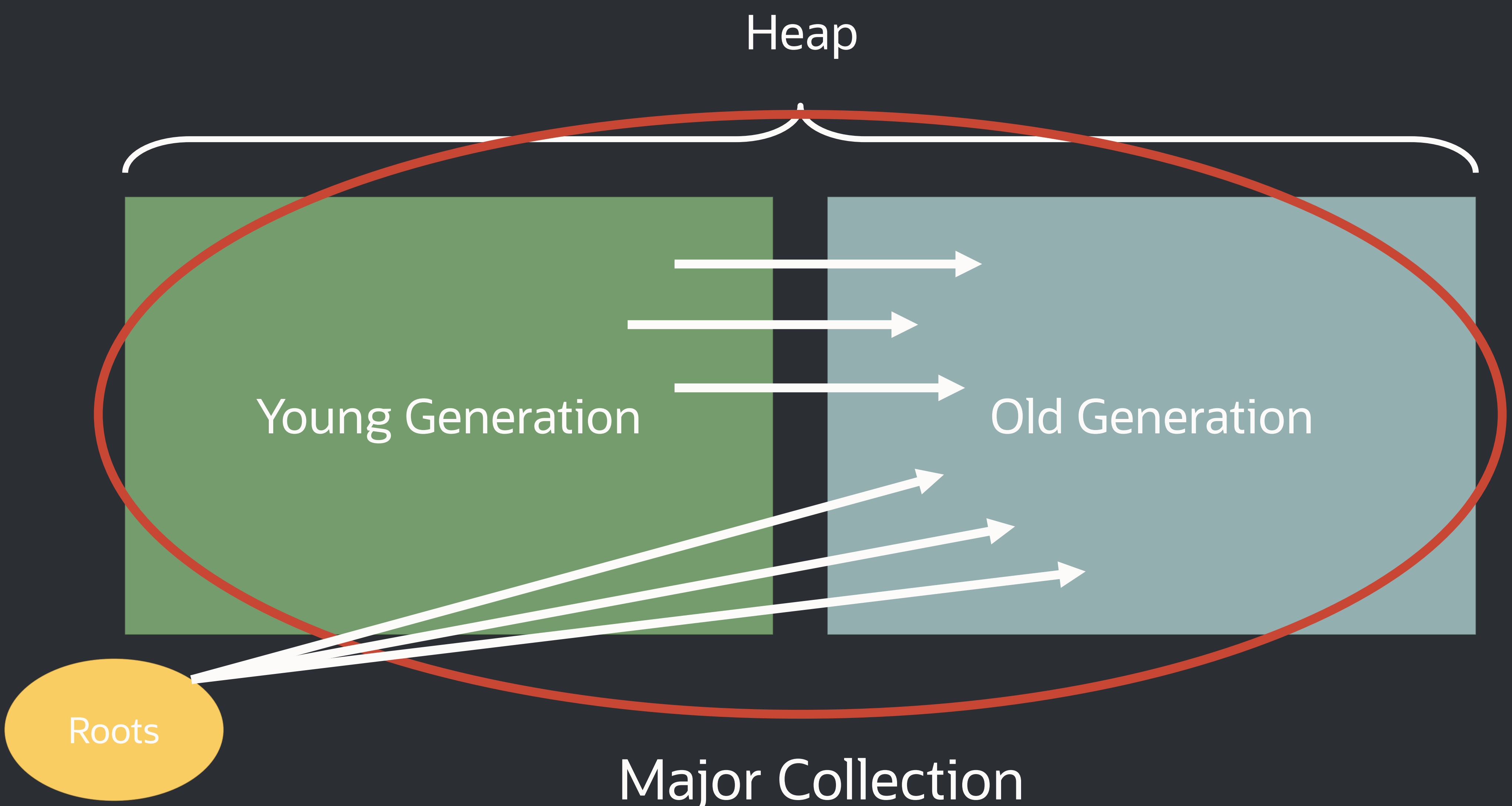
Major Collection

Copyright © 2022, Oracle and/or its affiliates



Major Collection

Copyright © 2022, Oracle and/or its affiliates



Major Collection

Copyright © 2022, Oracle and/or its affiliates

Automatic Tuning

Dynamic generation sizing

(No `–Xmn` needed)

Copyright © 2022, Oracle and/or its affiliates

Automatic Tuning

Dynamic tenuring threshold

(No `-XX:TenuringThreshold` needed)

Copyright © 2022, Oracle and/or its affiliates

Automatic Tuning

Dynamic number of threads

(No `-XX:ConcGCThreads` needed)

Copyright © 2022, Oracle and/or its affiliates

Automatic Tuning

Just set the max heap size!

(−Xmx)

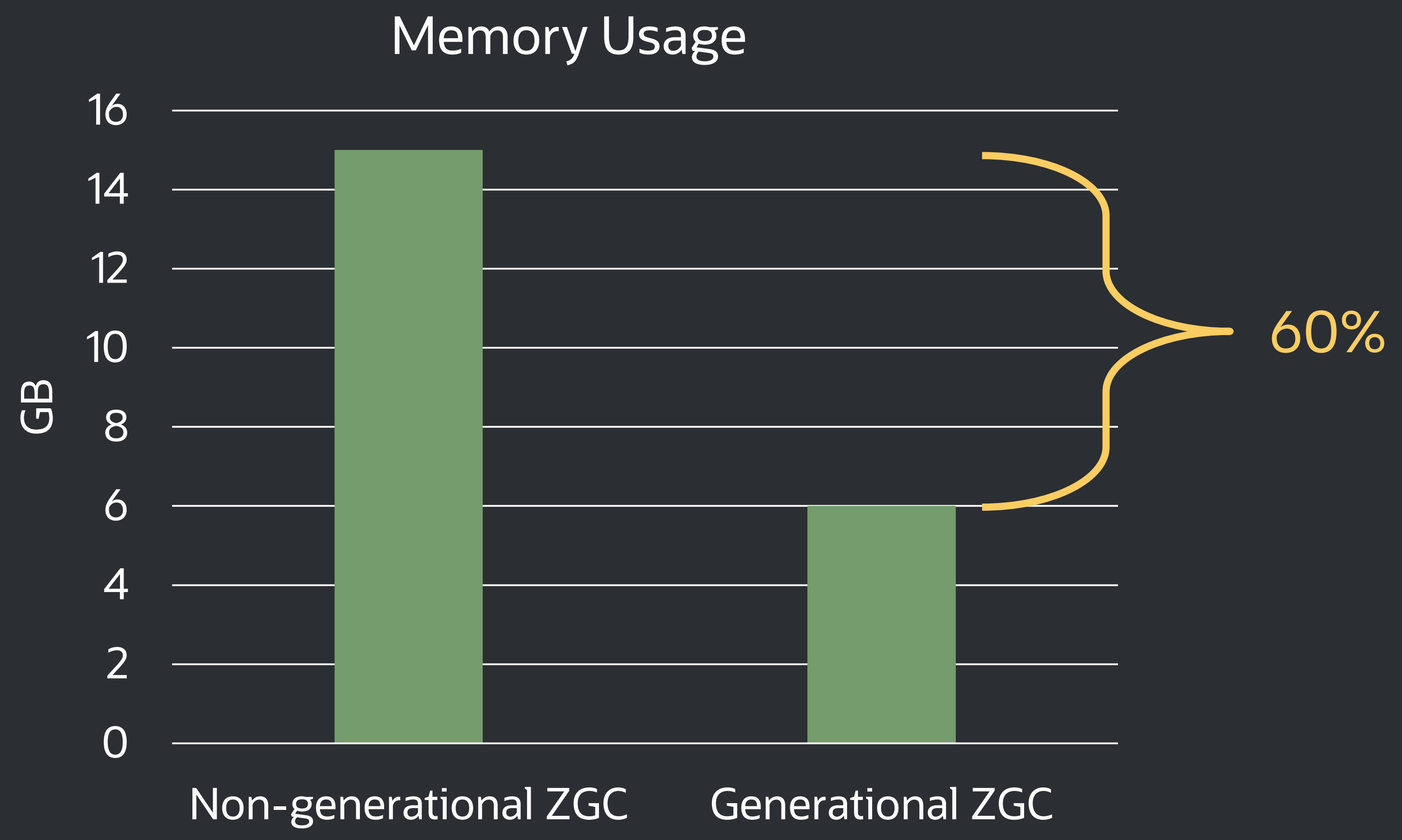
Copyright © 2022, Oracle and/or its affiliates

Preliminary Benchmarks

Copyright © 2022, Oracle and/or its affiliates

Memory Needed to Maintain Low Latency

Using the SPECjbb2005 benchmark with 1.5G live-set

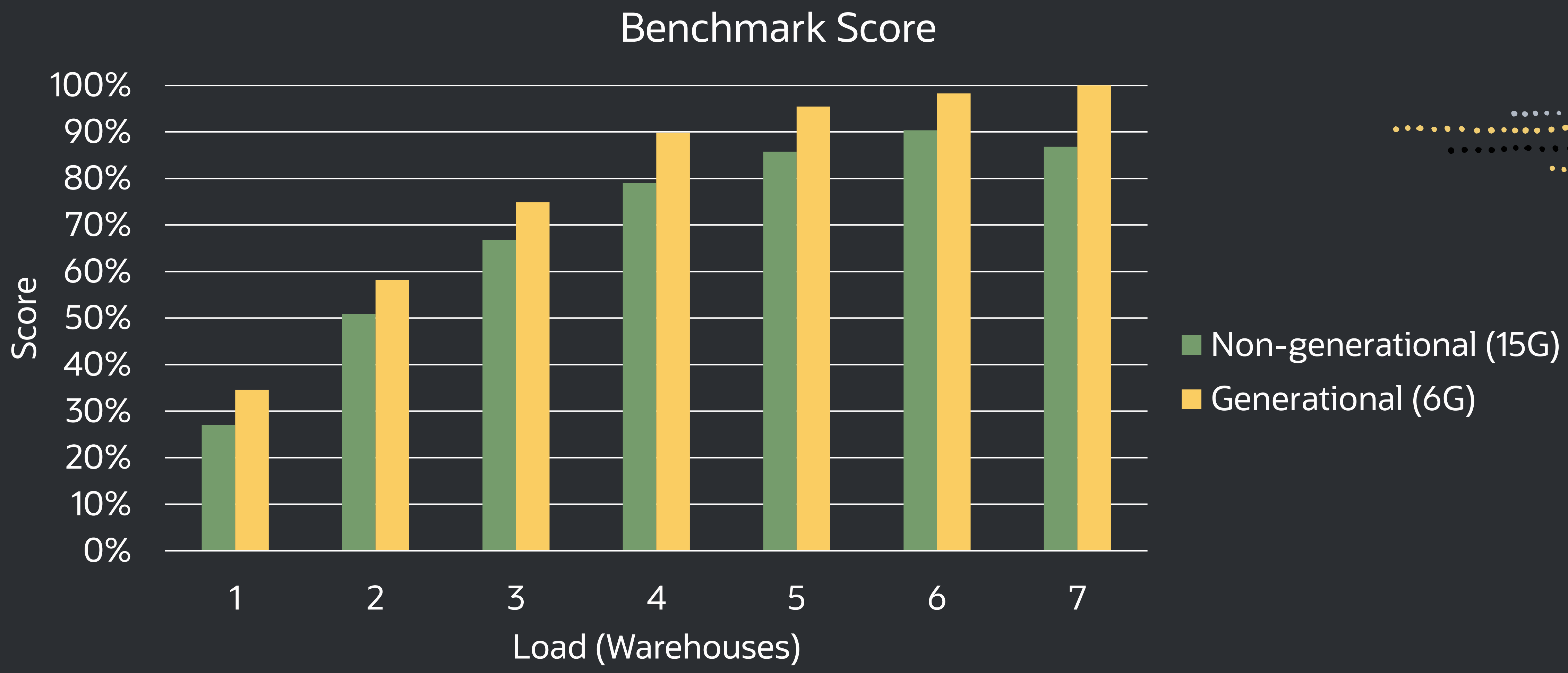


Copyright © 2022, Oracle and/or its affiliates

SPECjbb®2005 is a registered trademark of the Standard Performance Evaluation Corporation (spec.org). The actual results are not represented as compliant because the SUT may not meet SPEC's requirements for general availability.

Benchmark Score

Using the SPECjbb2005 benchmark with 1.5G live-set

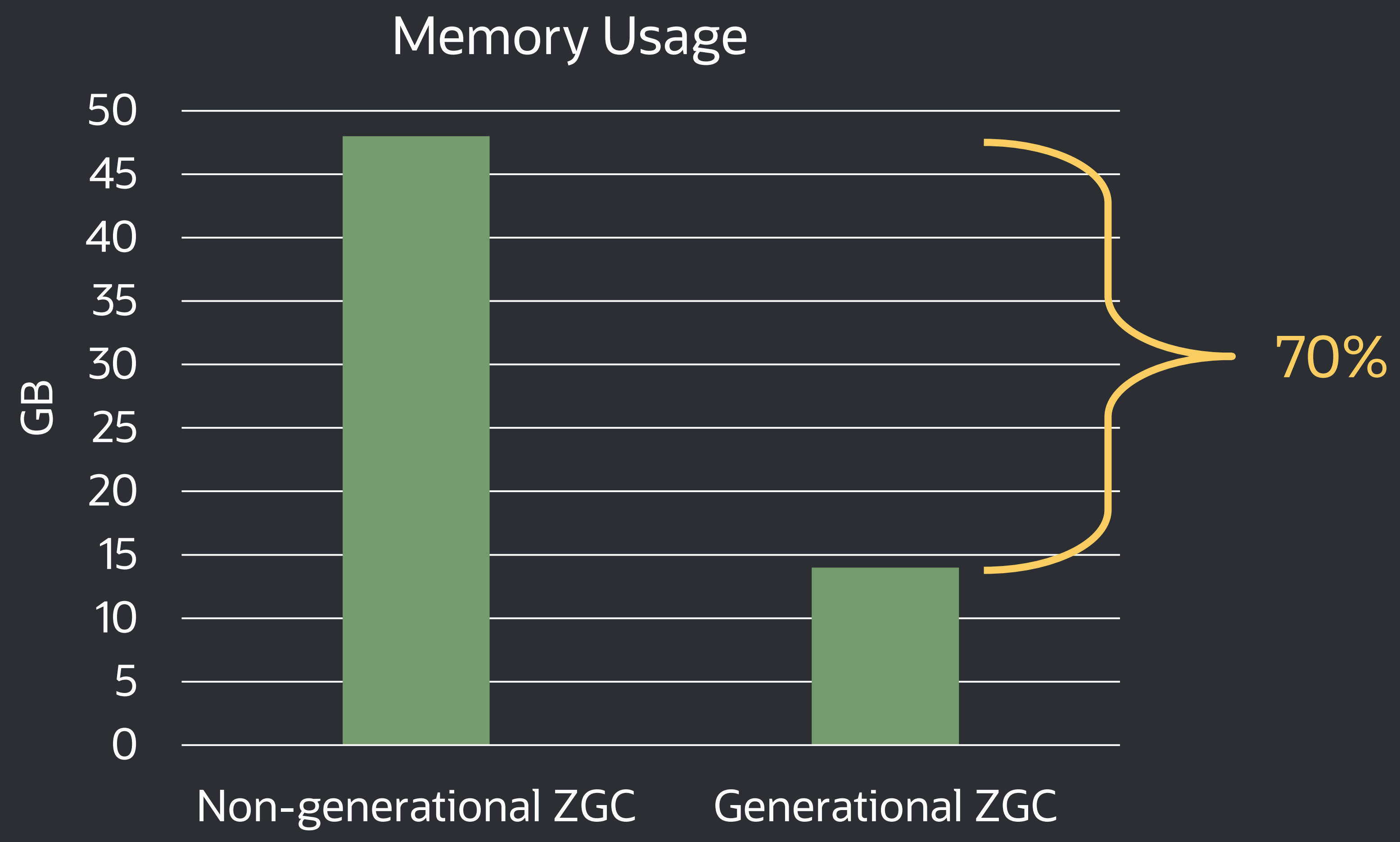


Copyright © 2022, Oracle and/or its affiliates

SPECjbb®2005 is a registered trademark of the Standard Performance Evaluation Corporation (spec.org). The actual results are not represented as compliant because the SUT may not meet SPEC's requirements for general availability.

Memory Needed to Maintain Low Latency

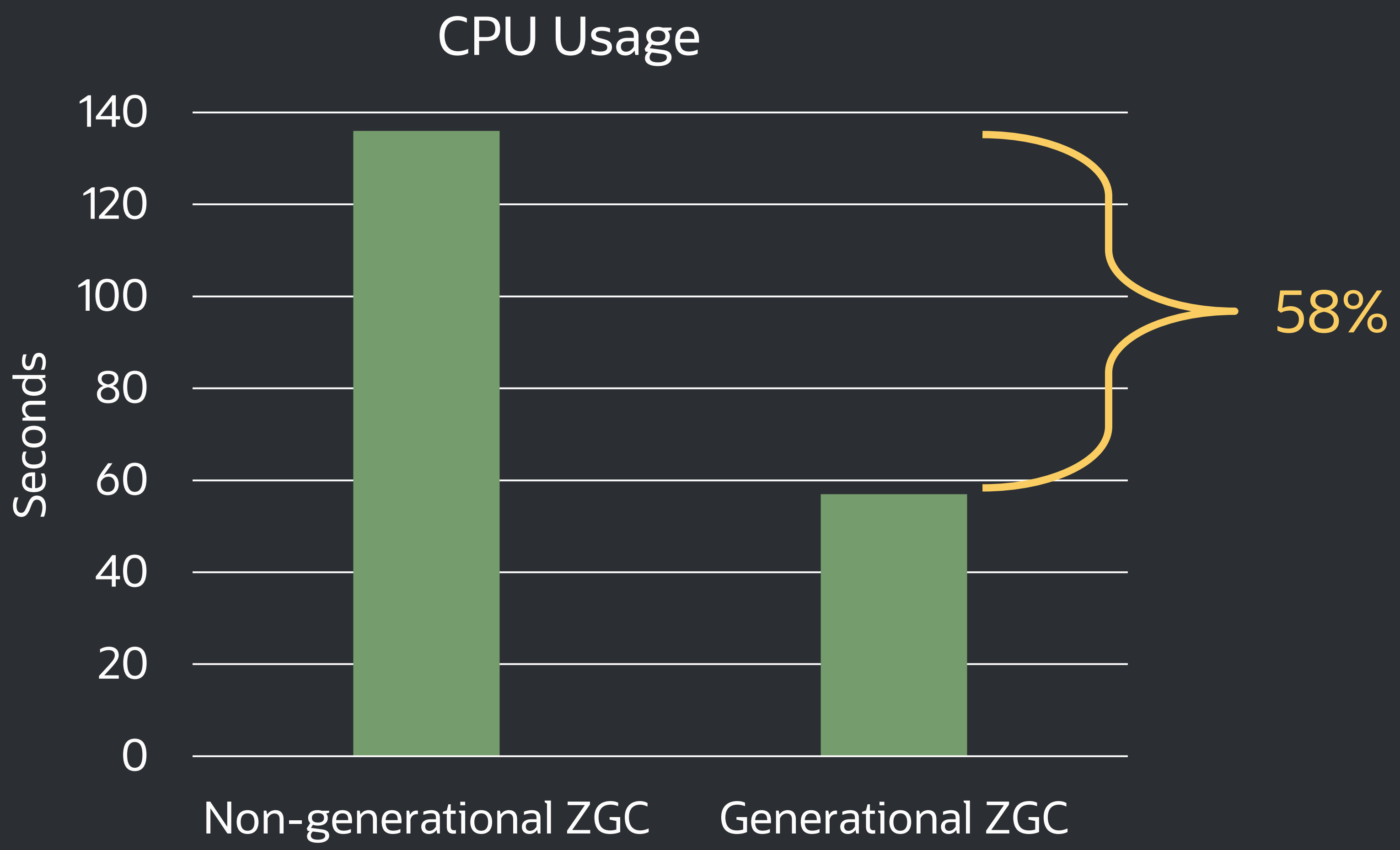
Using the Extremem benchmark



Copyright © 2022, Oracle and/or its affiliates

CPU Usage When Given the Same About of Memory

Using the Extremem benchmark

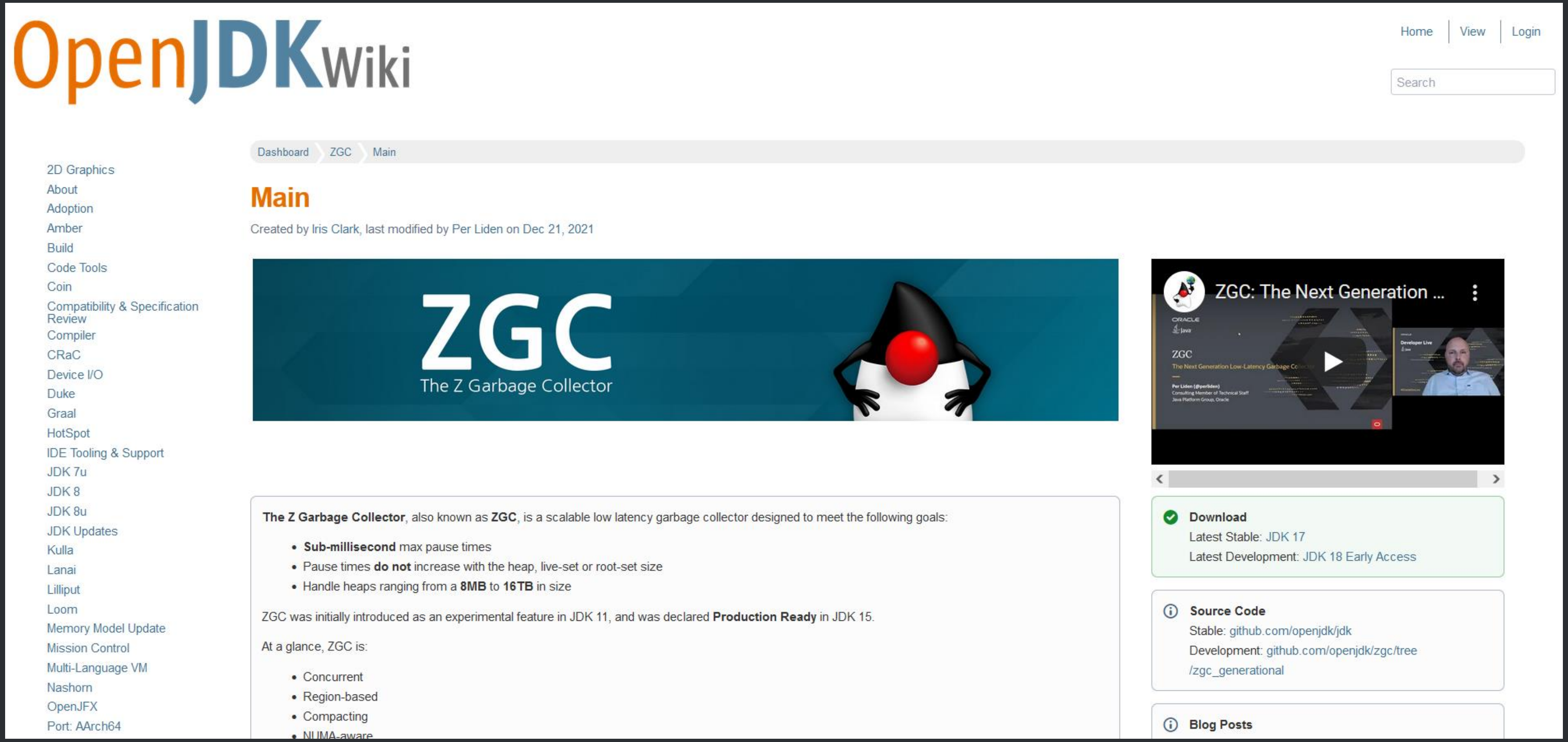


Copyright © 2022, Oracle and/or its affiliates

More Information

Copyright © 2022, Oracle and/or its affiliates

ZGC Wiki



Main

Created by Iris Clark, last modified by Per Liden on Dec 21, 2021

The Z Garbage Collector, also known as **ZGC**, is a scalable low latency garbage collector designed to meet the following goals:

- **Sub-millisecond** max pause times
- Pause times **do not** increase with the heap, live-set or root-set size
- Handle heaps ranging from a **8MB** to **16TB** in size

ZGC was initially introduced as an experimental feature in JDK 11, and was declared **Production Ready** in JDK 15.

At a glance, ZGC is:

- Concurrent
- Region-based
- Compacting
- NUMA-aware

Download
Latest Stable: JDK 17
Latest Development: JDK 18 Early Access

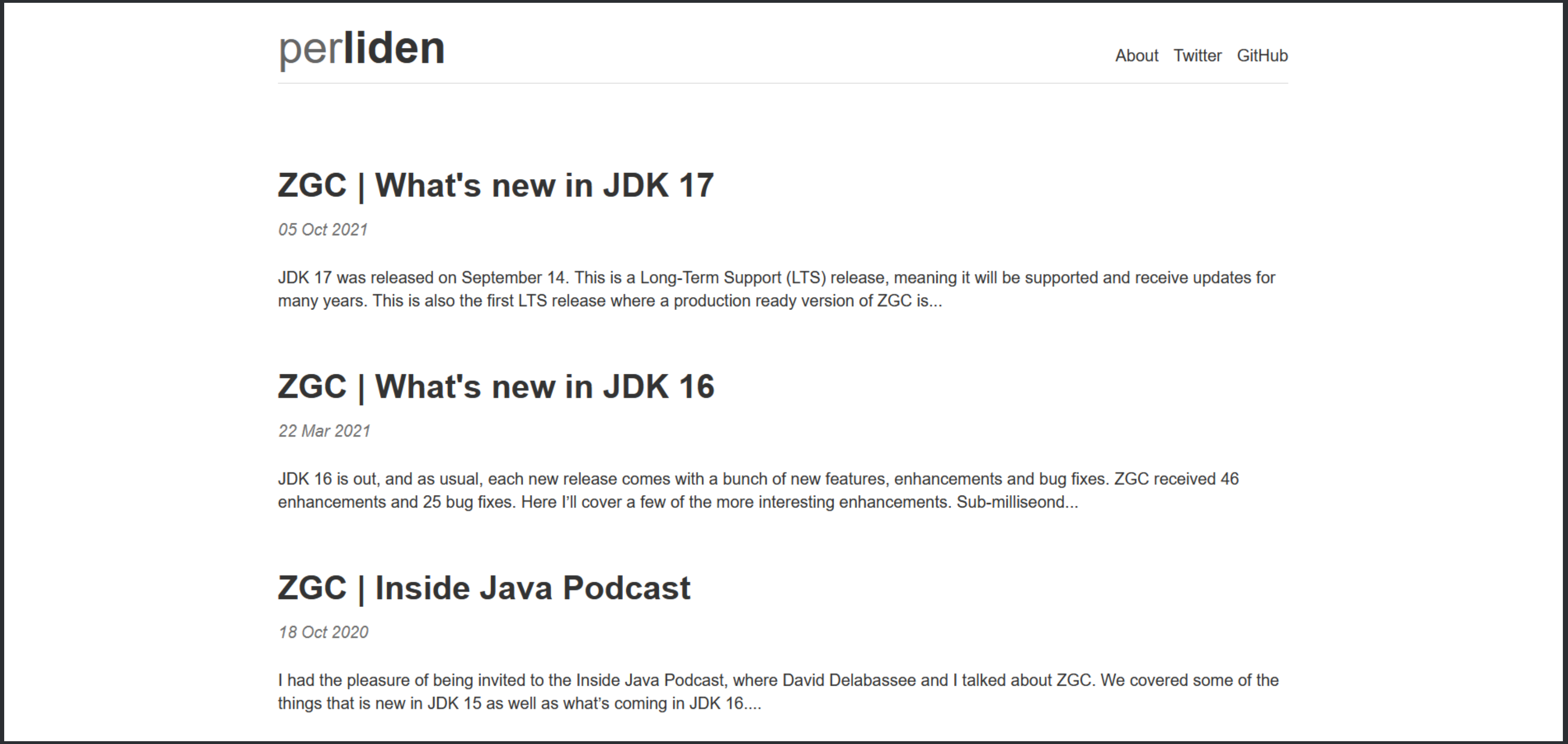
Source Code
Stable: github.com/openjdk/jdk
Development: github.com/openjdk/zgc/tree/zgc_generational

Blog Posts

https://wiki.openjdk.java.net/display/zgc

Copyright © 2022, Oracle and/or its affiliates

Blog



perliden

ZGC | What's new in JDK 17

05 Oct 2021

JDK 17 was released on September 14. This is a Long-Term Support (LTS) release, meaning it will be supported and receive updates for many years. This is also the first LTS release where a production ready version of ZGC is...

ZGC | What's new in JDK 16

22 Mar 2021

JDK 16 is out, and as usual, each new release comes with a bunch of new features, enhancements and bug fixes. ZGC received 46 enhancements and 25 bug fixes. Here I'll cover a few of the more interesting enhancements. Sub-milliseond...

ZGC | Inside Java Podcast

18 Oct 2020

I had the pleasure of being invited to the Inside Java Podcast, where David Delabassee and I talked about ZGC. We covered some of the things that is new in JDK 15 as well as what's coming in JDK 16....

https://malloc.se

Copyright © 2022, Oracle and/or its affiliates

Thanks!

Per Liden
E-mail: per.liden@oracle.com
Twitter: @perliden
Blog: malloc.se

Copyright © 2022, Oracle and/or its affiliates

Java™

Download Java 18 Now

oracle.com/javadownload

#OracleDevLive