JVM internals page 2 CHEAT SHEET

definitions

*The **heap** is the run-time data area from which memory for all class instances and arrays is allocated. Heap storage for objects is reclaimed by an automatic storage management system (known as a **garbage collector**).*

***HotSpot** JVM distinguishes the following memory areas:*

***Eden space** is a memory pool in which memory is initially allocated.*

***Survivor space** is a memory pool which contains objects that survived garbage collection in Eden space.*

***Young generation** consists of eden and two equally sized survivor spaces.*

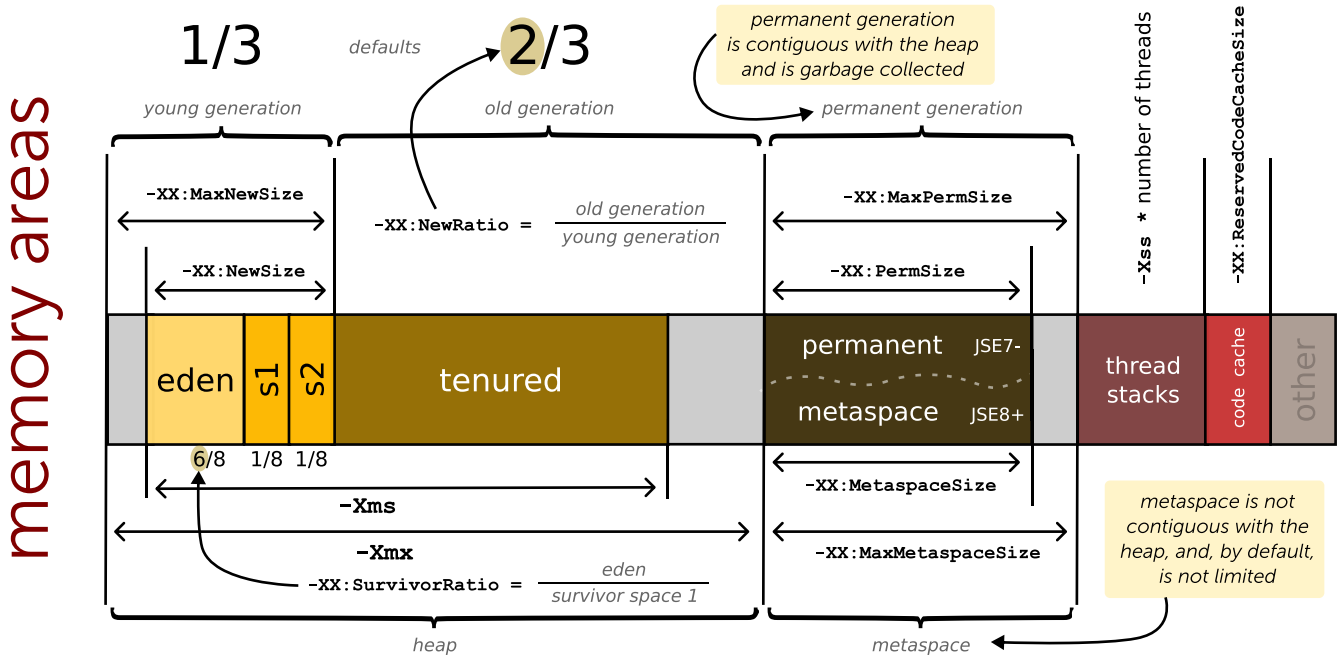
***Tenured (old) generation** contains objects that passed several garbage collections in the survivor spaces.*

***Permanent generation** contains class and method data. It is NOT part of the heap, but is contiguous with the heap. It was replaced with metaspace in JDK 8.*

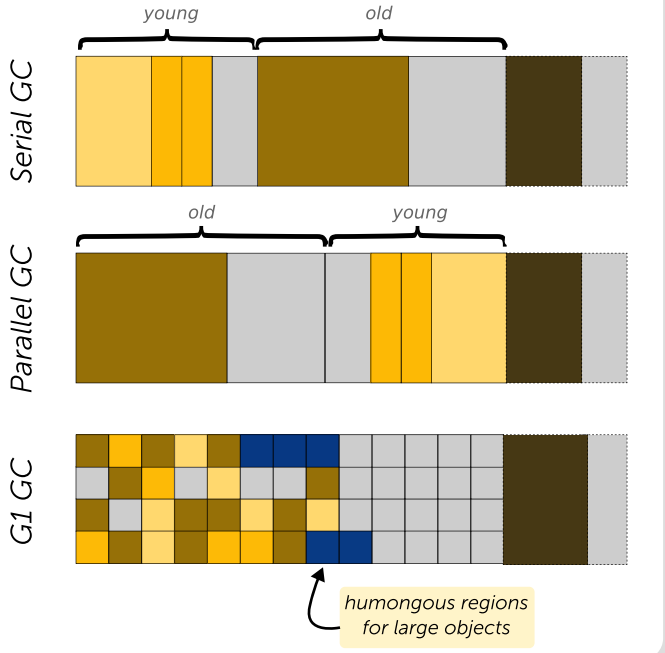
***Metaspace** is not contiguous with the heap and has different management mechanics. It is not limited by default, and, therefore, theoretically, can fill all available OS memory.*

***Code cache** contains native code compiled by **JIT** compiler at runtime.*

memory areas



memory layout



collectors

*A **serial collector** uses a single CPU core to perform garbage collection.*

*A **parallel collector** uses multiple threads that can perform collection work at the same time, and can use of more than one CPU core.*

*A **stop-the-world (STW) collector** performs garbage collection during a pause when the application is not running any code that could mutate the heap.*

*A **concurrent collector** performs garbage collection concurrently with program execution, i.e. allowing program execution to proceed whilst collection is carried out.*

extreme automation