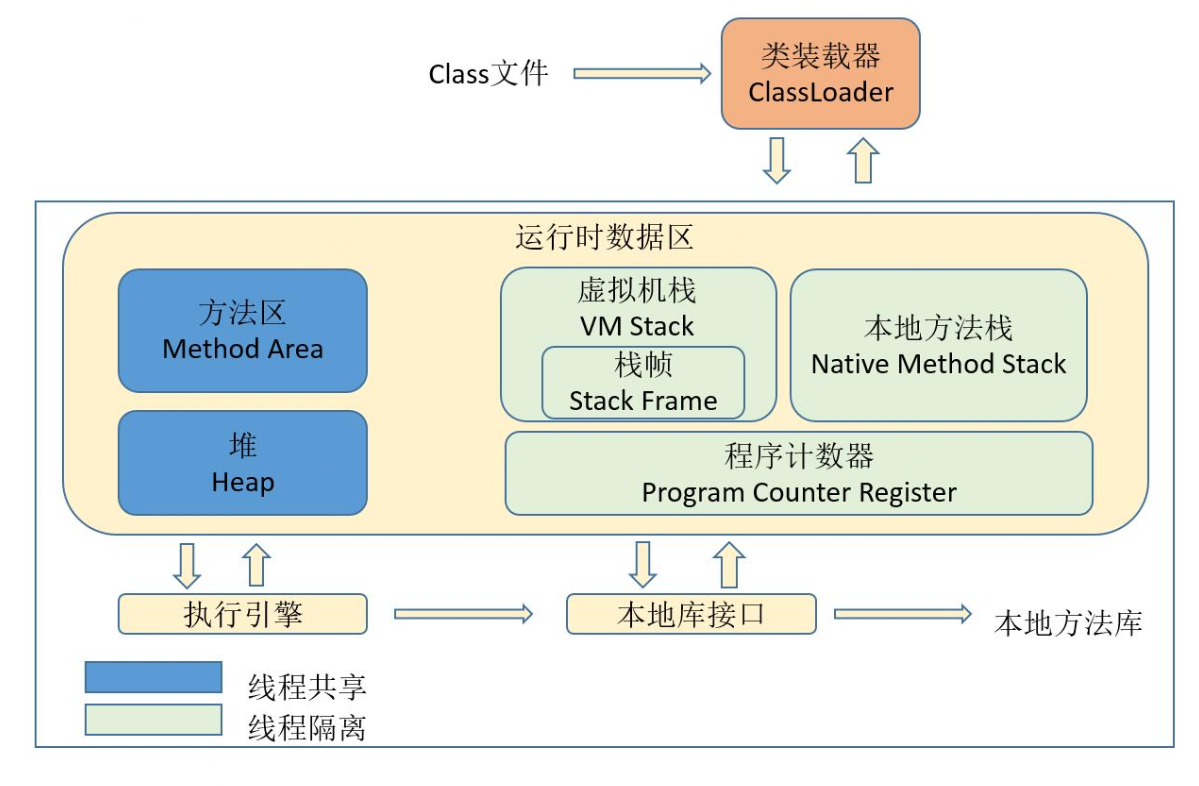
**深入理解JVM**

**Part 1 JAVA内存区域与内存溢出异常**

1. 程序计数器

当前正在执行的指令的地址，如果调用的Native方法，则是Undefined。

1. JAVA虚拟机栈

内部由栈帧组成，一个栈帧代表一个调用的方法。

1. 本地方法栈

由于很多 Native 方法都是用 C 语言实现的，所以它通常又叫 C 栈。

1. 为虚拟机使用到的本地方法服务
2. JAVA堆

存放对象和数组，GC堆。可以在物理上不连续，逻辑连续即可。

1. 方法区

用于存储已被虚拟机加载过的类信息，常量，静态变量。

**PART 2 垃圾收集器与内存分配策略**

**2.1可达性分析算法**

如果对象与GC Roots没有任何关联，则视为该对象不可用。

GC管理的主要区域是堆内存（不包括方法区），所以虚拟机栈，本地方法栈，方法区这些区域不被GC所管理，因而选择这些区域的对象作为GC Roots。

GC Roots 的对象包括一下几种类型：

（1）虚拟机栈（栈帧中的本地变量表）中引用的对象。

（2）方法区中类静态属性引用的对象。

（3）方法区中的常量引用的对象。

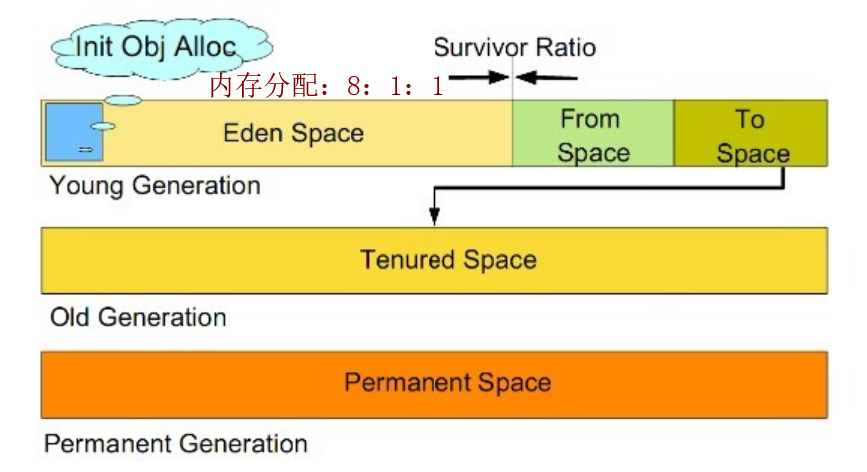
（4）本地方法栈中JNI（即一般说的Native方法）引用的对象。

例：Class Thread Stack Local JNI Local

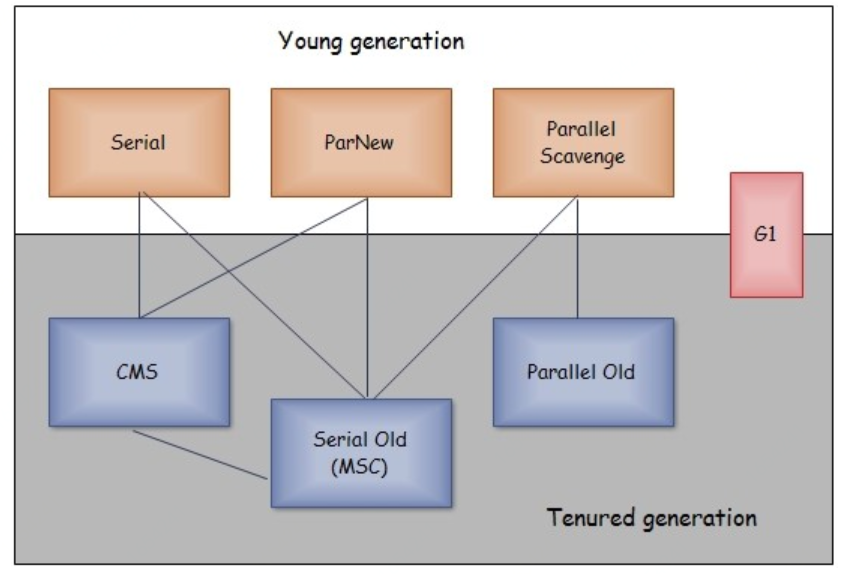
**2.2垃圾收集算法**

老年代：标记-清除算法（容易产生大量碎片），标记-整理算法

新生代：复制算法 【如果另一块survivor区域内存不足，则会使用老年代的内存---分配担保机制】



**2.3垃圾收集器**



【补】JVM中的 server模式与client模式区别

**2.4内存分配与回收策略**