.NET垃圾回收

.Net的垃圾回收机制使用标记清除整理(mark-sweep-compacting)算法。.Net引入了根（强应用，相对还有弱引用）的概念，根一般是全局变量和静态的变量。通过 根的可达判断，来标识那些对象将被垃圾收集器回收。同时，.Net还引入对象的代龄和安全点来优化垃圾回收机制的性能。

**.Net垃圾回收机制中几个重要的概念:**

* 垃圾回收算法的选择
* 对象的代龄及线程安全点
* Finalize操作和Dispose模式

**垃圾回收算法的选择**

**对象的代龄及线程安全点**

代龄是旨在提高垃圾收集器性能的一种机制，它假设

* 对象越新，其生存期越短。
* 对象越老，其生存期越长。
* 对托管堆的一部分执行垃圾收集要比对整个托管堆执行垃圾收集速度更快。

CLR的托管堆支持3个代龄，每个代龄都有相应的阀值容量，并在垃圾回收的过程中进行调整。当相应代龄被填满后，垃圾收集器启动。此后.Net将从下一个代龄开始分配对象。

线程安全点是在垃圾收集执行完毕之前一直挂起线程的地方。

**Finalize操作和Dispose模式**

Finalize相当于C++中类的析构函数，它是给垃圾收集器被动调用的。常见于第0代对象充满的时候。

Dispose是显式调用的。