Atitit gc 垃圾回收原理与概论and 自动资源管理的艺术

[1.1. 自动资源管理的对象db conn,http conn,file 1](#_Toc31302)

[2. 如何关联----wrap file，network socket 1](#_Toc28650)

[2.1. 垃圾回收的算法与实现 Ruby之父松本行弘推荐GC算法 Garbage Collection 一本书掌握自动内存回收的机制 2](#_Toc16257)

[2.2. 垃圾回收算法手册：自动内存管理的艺术 3](#_Toc14243)

## 自动资源管理的对象db conn,http conn,file

# 如何关联----wrap file，network socket

我希望改进这一点，也就是说，那所有 gc 相关的数据集中在一起，整个收集过程，除了最终释放那些不再使用的内存外，不会碰用户数据块的内存。

gc 最重要的一点，就是要对堆栈上的数据进行关联。在收集发生时，堆栈上所有临时分配出来的内存块都不应该被释放掉。C 语言本身不提供堆栈遍历的特性，所以要想个自然的方案让用户可以方便的做到这点。

在用户的调用栈上，每个调用级上，临时分配的内存都被自然挂接在当前级别的堆栈挂接点上，一旦调用返回，当前级别的所有临时内存块都应该和根断开。当然，如果内存块作为返回值出现的话，需要保留。在 C 里，我们需要给每个函数的入口和出口都做一个监护，保证 gc 的正确工作。（如果是 C++ ，要稍微方便一点，在函数进入点设置一个 guard 对象即可）因为这个监护过程会非常频繁，对其的优化是重点工作。

Atitit gc资源释放器死锁解除器 适合于游戏引擎，数据库释放 爬虫网络连接释放等

## 垃圾回收的算法与实现 Ruby之父松本行弘推荐GC算法 Garbage Collection 一本书掌握自动内存回收的机制

第1章　学习GC之前

第2章　GC标记-清除算法

第3章　引用计数法

第4章　GC复制算法

第5章　GC标记-压缩算法

第6章　保守式GC

第7章　分代垃圾回收

第8章　增量式垃圾回收

第9章　RC Immix算法

第10章　Python的垃圾回收

10.4　第0层 通用的基础分配器　197

10.5　第1层 Python低级内存分配器　198

10.6　第2层 Python对象分配器　208

10.7　第3层 对象特有的分配器　231

第11章　DalvikVM的垃圾回收

第12章　Rubinius的垃圾回收

第13章　V8的垃圾回收

## 垃圾回收算法手册：自动内存管理的艺术

第1章　引言 1

第2章　标记–清扫回收 15

第3章　标记–整理回收 28

第4章　复制式回收 38

第5章　引用计数 49

第6章　垃圾回收器的比较 64

第7章　内存分配 72

第8章　堆内存的划分 87

第9章　分代垃圾回收 95

第10章　其他分区策略 117

第11章　运行时接口 138

第12章　特定语言相关内容 188

第13章　并发算法预备知识 202

第14章　并行垃圾回收 242

第15章　并发垃圾回收 271

第16章　并发标记–清扫算法 285

第17章　并发复制、并发整理算法 298

第18章　并发引用计数算法 322

第19章　实时垃圾回收 333

《垃圾回收算法手册：自动内存管理的艺术》([英] 理查德·琼斯)【简介\_书评\_在线阅读】 - 当当图书.html