



引用计数

进入词条

全站搜索

帮助

声明：百科词条人人可编辑，词条创建和修改均免费，绝不存在官方及代理商付费代编，请勿上当受骗。 [详情>>](#)

首页

秒懂百科

特色百科

用户

知识专题

权威合作

下载百科APP

个人中心

# 引用计数

播报

编辑

讨论

上传视频



收藏



32



1

本词条由“[科普中国](#)”科学百科词条编写与应用工作项目 审核。

引用计数是计算机编程语言中的一种[内存管理技术](#)，是指将资源（可以是[对象](#)、[内存](#)或[磁盘空间](#)等等）的被[引用](#)次数保存起来，当被引用次数变为零时就将其释放的过程。使用引用计数技术可以实现自动资源管理的目的。同时引用计数还可以指使用引用计数技术回收未使用资源的[垃圾回收算法](#)。

中文名	引用计数	类 型	最直观的垃圾收集策略
定 义	<a href="#">内存管理技术</a>	学 科	<a href="#">计算机</a>
相 关	<a href="#">垃圾回收算法</a>	领 域	<a href="#">计算机编程语言</a>

目录	1 简介	▪ 规则	▪ 规则
	2 引用计数的使用	▪ 接口	3 优势与缺陷
	▪ 原因	▪ 调试	

## 简介

播报

编辑

最直观的垃圾收集策略是引用计数。引用计数很简单，但是需要[编译器](#)的重要配合，并且增加了[赋值函数](#) (mutator) 的开销（这个术语是针对[用户程序](#)的，是从垃圾收集器的角度来看的）。每一个对象都有一个关联的引用计数 —— 对该对象的活跃引用的数量。如果对象的引用计数是零，那么它就是垃圾（用户程序不可到达它），并可以回收。每次修改[指针](#)引用时（比如通过[赋值语句](#)），或者当引用超出范围时，编译器必须生成代码以更新引用的对象的引用计数。如果对象的引用计数变为零，那么运行时就可以立即收回这个块（并且减少被回收的块所引用的所有块的引用计数），或者将它放到延迟收集队列中。

[com组件](#)将维护一个称作是引用计数的数值。当客户从组件取得一个接口时，此引用计数值将增1。当客户使用完某个接口后，组件的引用计数值将减1.当引用计数值为0时，组件即可将自己从内存中删除。

## 引用计数的使用

播报

编辑

### 原因

为什么要选择为每一个接口单独维护一个引用计数而不是针对整个组件维护引用计数呢？

主要有两个原因：一是使[程序调试](#)更为方便；另外一个原因是支持资源的按需获取。

#### 1.程序调试：

假设在程序中忘记对某个接口调用Release（其实很多人会犯这个错）。这样组件将永远不会被删除掉，因为只是在引用计数0时delete才会被调用。这时就需要找出接口在何时何处应该被释放掉。当然找起来是相当困难的。在只对整个组件维护一个接口的情况下，进行这种查找更为困难了。此时必须检查使用了此组件所提供的所有接口的代码。但若组件支持对每个接口分别维护一个引用计数那么可以把查找的范围限制在某个特定的接口上。在某些情况下这可以节省大量时间。

#### 2.资源的按需获取

在实现某个接口时可能需要大量的内存或其他资源。对于此种情况，可以在[QueryInterface](#)的实现中，在客户请求此接口时完成资源的分配。但若只对整个组件维护一个引用计数，组件将无法决定何时可以安全地将这些接口相关联的[内存释放](#)。但基对每个接口分别维护一个引用计数，那么决定何时可以将此内存释放将会容易得多。

### 规则

#### 正确使用引用计数三条简单的规则

1. 在返回之前调用AddRef。对于那些建好些返回接口[指针](#)的函数，在返回之前应该相应的指针调用AddRef。这些函数包括QueryInterface 及CreateInstance。这样当客户从这种函数得到一个接口后。它将无需调用AddRef。

### 科普中国

致力于权威的科学传播

本词条认证专家为

王冲 | 副教授、副研究员  
中国科学院工程热物理研究所

V百科

往期




### 心有绿意

分享「植物之美」



权威合作编辑



“科普中国”科学百科

“科普中国”是为我国科普信息化建设塑造的全...

什么是权威编辑

#### 词条统计

浏览次数：40033次  
编辑次数：15次[历史版本](#)  
最近更新：你饿嘛（2021-07-26）

#### 突出贡献榜

mexia 

- 1 哪个电竞学校好

2 孩子弱智的表现

3 胸缩小了怎么办

4 情侣网站

5 怎么上哈佛大学

6 自学编程入门

7 病娇心理测试题

8 接触角测量仪器

9 自学编程

10 美国私人vps

11 学会日语

12 自动计数包装机
- 13 土石方车辆计

14 双室真空包装

15 网站入口

16 姓名爱情测试

17 电子秤多少钱

18 什么叫做助孕

19 抹胸是什么

20 显卡天梯图

陈铭

微光演讲纯享

演讲技巧

演讲技巧



2.使用完接口之后调用Release。在使用某个接口之后应该调用些接口的Release函数。

3.在赋值之后调用AddRef. 将一个接口指针赋给另一个接口指针时，应调用AddRef。换句话说，在建立接口的别外一个引用之后应增加相应组件的引用计数。

## 接口

在客户看来，引用计数是处于接口级的而不是组件级的。但从实现的角度来看，谁的引用计数被记录下来实际上没有关系。客户可以一直接相信组件将记录每个接口本身维护引用计数值。但客户不能假设整个组件维护单个的引用计数。

对于客户而言，每一个接口被分别维护一个引用计数意味着客户应该对它将要使用的指针调用AddRef，而不是其他的什么指针。对于使用完了指针客户应该调用其Release。

选择为每一个接口单独维护一个引用计数而不是针对整个组件维护引用计数的原因:

使程序调试更为方便；支持资源的按需获取；

## 调试

可以通过增大和减少某个数的值而实现之。

另外要注意的是AddRef和Release的返回值没有什么意义，只是在程序调试中才可能会用得上.客户不应将此从此值当成是组件或其接口的精确引用数。

## 规则

客户必须对每一个接口具有一个单独的引用计数值那样来处理各接口。因此，客户必须对不同的接口分别进行引用计数，即使它们的生命期是嵌套的。

### 一、输出参数规则

输出参数指的是给函数的调用者传回一个值的函数参数。从这一点上讲，输出参数的作用同函数的返回值是类似的。任何在输出参数中或作为返回值返回一个新的接口指针的函数必须对这些接口指针调用AddRer。

### 二、输入参数规则

对传入函数的接口指针，无需调用AddRef和Release，这是因为函数的生命期嵌套在调用者的生命期内。

### 三、输入－输出参数规则

输入－输出参数同时具有输入参数及输出参数的功能。在函数休中可以使用输入－输出参数的值，然后可以对这些值进行修改并将其返回给调用者。

在函数中，对于用输入－输出参数传递进来的接口指针，必须在给它赋另外一个接口指针值之前调用其Release。在函数返回之前，还必须对输出参数中所保存的接口指针调用AddRef。

### 四、局部变量规则

对于局部自制的接口指针，由于它们只是在函数的生命其内才存在，因此无需调用AddRef和Release。这条规则实际是输入参数规则的直接结果。在下面的例子中，piX2只是在函数foo的生命期内都在，因此可以保证其生命期将嵌套在所传入的piX指针的生命期，因此无需对piX2调用AddRef和Release。

### 五、全局变量规则

对于保存在全局变量中的接口指针，在将其传递给另外一个函数之前，必须调用其AddRef。由于此变量是全局性的，因此任何函数都可以通过调用其Release来终止其生命期。对于保存在成员变量中的接口指针，也应按此种方式进行处理。因为类中的任何成员函数都可以改变此种接口指针的状态。

### 六、不能确定时的规则

对于任何不定的情形，都应调用AddRef和Release对。

另外，在决定要进行优化时，应给那些没有进行引用计数的指针加上相应的注释，否则其它程序员在修改代码时，将可能会增大接口指针的生命期，从而合引用计数的优化遭到破坏。

忘记调用Release造成的错误可能比不调用AddRef造成的错误更难检测。

## 优势与缺陷

🔊 播报    ✎ 编辑

与跟踪式垃圾回收相比，引用计数的主要优点是可以尽快地回收不再被使用的对象，同时在回收过程中不会导致长时间的停顿，还可以清晰地标明每一个对象的生存周期。

在实时应用或内存受限的系统中，实时响应能力是一项重要指标，而引用计数作为最容易实现的垃圾回收技术之一，很适合于这种情况。引用计数还可以用于管理其他非内存资源，如操作系统对象（经常比内存资源更稀缺）。跟踪式垃圾回收技术用终结器处理此类目标，但延迟回收可能引发其他问题。加权引用计数是适用于分布式系统的派生技术。



陈铭  
微光演讲纯享



在可用内存被活跃对象填满的平台上，跟踪式垃圾回收会被频繁触发，从而降低性能。而引用计数即便在内存濒临耗尽的情况下性能依然有所保障。引用计数还能为其他运行时优化技术提供参考信息，例如对于许多使用[不可变对象](#)的系统来说（如函数式编程语言），大量复制对象导致的性能惩罚有时十分严重；在此类系统上一个典型的优化措施是：假如一个对象被创建以后仅使用了一次，且在其不再被引用的同时另一个类似的对象被创建出来（如[Javascript](#)中的字符串拼接赋值操作），可以将删除原对象创建新对象的行为变为修改原对象，从而提高效率。引用计数可以为这类优化提供充分的参考信息。<sup>[\[1\]](#)</sup>

未经优化的引用计数相比跟踪式垃圾回收有两个主要缺点，都需要引入附加机制予以修复：

- 频繁更新引用计数会降低运行效率。
- 原始的引用计数无法解决[循环引用](#)问题。

另外，如果使用空闲列表分配内存，那么引用计数的空间局域性非常差。仅使用引用计数无法通过移动对象来提高[CPU缓存](#)的性能，所以高性能的内存分配器都会同时实现一个跟踪式垃圾回收器以提高性能。许多引用计数实现（比如PHP和Objective-C）的性能不佳都是因为没有实现内存拷贝。<sup>[\[2\]](#)</sup>

### 参考资料

1.

Wilson, Paul R. "Uniprocessor Garbage Collection Techniques". Proceedings of the International Workshop on Memory Management. London, UK: Springer-Verlag. pp. 1–42. ISBN 3-540-55940-X. Retrieved 5 December 2009. Section 2.1.

2.

Rifat Shahriyar, Stephen M. Blackburn, Xi Yang and Kathryn S. McKinley (2013). "Taking Off the Gloves with Reference Counting Immix". 24th ACM SIGPLAN conference on Object Oriented Programming Systems, Languages and Applications. OOPSLA 2013

### 搜索发现

- [什么是引用计数](#)
- [化妆学校前十名](#)
- [三叶草官网](#)
- [标致408降价](#)
- [沙盘模型](#)
- [股票推荐](#)
- [玉器批发](#)
- [小户型](#)
- [襄阳同城游戏](#)
- [ps基础入门教程](#)

### 新手上路

- [成长任务](#)
- [编辑入门](#)
- [编辑规则](#)
- [本人编辑](#) **NEW**

### 我有疑问

- [内容质疑](#)
- [在线客服](#)
- [官方贴吧](#)
- [意见反馈](#)

### 投诉建议

- [举报不良信息](#)
- [未通过词条申诉](#)
- [投诉侵权信息](#)
- [封禁查询与解封](#)

