**用google-perftool分析程序的内存/CPU使用**

2012年1月3日[小武哥](http://www.wuzesheng.com/?author=2)[发表评论](http://www.wuzesheng.com/?p=2147#respond)[阅读评论](http://www.wuzesheng.com/?p=2147#comments)

最近，用到了google-perftool分析程序的内存和CPU的使用情况，总结一下使用的一些方法和体会，分享给有需要的朋友。首先，说说google-perftool，它是由google开发的用来分析C/C++程序性能的一套工具，这里的性能分析主要包括内存和CPU两个方面，内存分析使用google-perftool所提供的tcmalloc，CPU分析使用它所提供的profiler。下面先分别介绍一下tcmalloc和profiler，然后再给出一些使用的例子，及一些使用时的注意事项。

 1. tcmalloc

tcmalloc的全称是thread cache malloc，顾名思义，它是带有thread cache的内存管理工具，具体的实现细节这里不做过多的介绍，感兴趣的朋友可以参考google官方提供的文档，或者阅读源码。这里需要注明一下tcmalloc的一些优点，和它所提供的一些分析程序内存使用的一些功能。  
tcmalloc的主要优点有两个方面，一个是内存allocate/deallocate的速度，通常情况下它的速度比glibc所提供的malloc要快；另一个方面是小内存(< =32K)的管理，它的小内存是在thread cache里面管理的，一方面减少了加锁的开销，另一方面用来表示小内存所用的额外的空间也比较小，比较节省空间。因此，对于多线程下，经常小内存的allocation/deallocation的程序（尤其多线程下使用STL比较多的程序），可以尝试使用一下tcmalloc。  
除了在allocate/deallocate内存时的优化外，tcmalloc还提供了heapcheck和heapprofile的功能。heapcheck主要被用来检查程序中是否有内存泄露，在哪里泄露。相信内存泄露这个话题，永远是让所有C/C++程序都非常蛋疼的问题，有了tcmalloc的帮助，也许一切会变得简单一些，会有点事半功倍的效果。tcmalloc另外一个功能是heapprofile。先来说一下profile这个词，它的本意是“描绘…轮廓”，我一直觉得这个词是一个很伟大的词，用在sns中，它表示用来描绘用户的那些特征、属性，也有叫用户画像的。用在这里呢，heapprofile，顾名思义，它就是描绘程序的heap轮廓，通过这样一个过程，我们就能知道，程序的heap里在每一时刻都有些啥东东。有了profile的结果，它可以帮助我们定位内存泄露，帮助我们发现一些频繁allocate内存的地方，以此来做一些优化。  
（关于程序中的heap和stack，不了解的朋友，可以到网上搜一下：程序的地址空间分布）

 2. profiler  
profiler，是由google-perftool所提供的用来做cpu-profile的工具，相信通过上面的介绍，大家对profile这个词已经不再陌生。Cpu-profile，它的主要功能就是通过采样的方式，给程序中cpu的使用情况进行“画像”，通过它所输出的结果，我们可以对程序中各个函数耗时情况一目了然。在对程序做性能优化的时候，这个是很重要的，先把最耗时的若干个操作优化好，程序的整体性能提升应该十分明显，这也是做性能优化的一个最为基本的原则—先优化最耗时的。

 3. 使用举例

关于google-perftool的使用，总体上来讲有以下三种方式：

* （1）直接调用提供的api：这种方式比较适用于对于程序的某个局部来做分析的情况，直接在要做分析的局部调用相关的api即可。
* （2）链接静态库：这种方式是最为常用的方式，后面会有详细的介绍。
* （3）链接动态库：这种方式和静态库的方式差不多，但通常不推荐使用，除非使用者不想额外链一个静态库（因为链接静态库会增大binary的大小）的情况，可以考虑使用这种方式。

下面是静态库的使用方法：

* （1）Heap Checker:

|  |  |
| --- | --- |
| 1  2 | gcc [...] -o myprogram -ltcmalloc  env HEAPCHECK=normal ./myprogram |

* （2）Heap Profiler:

|  |  |
| --- | --- |
| 1  2 | gcc [...] -o myprogram -ltcmalloc  env HEAPPROFILE=./myprogram.hprof ./myprogram |

* （3）Cpu Profiler:

|  |  |
| --- | --- |
| 1  2 | gcc [...] -o myprogram -lprofiler  env CPUPROFILE=./myprogram.prof ./myprogram |

通过上面的方法，可以生成google-perftool原始的输出结果。另外，google-perftoolg还提供了一个叫pprof的工具，它是一个perl的脚本，通过这个工具，可以将google-perftool的输出结果分析得更为直观，输出为图片、pdf等格式。

最后，补充一点，要用google-perftool来分析程序，必须保证程序能正常退出。所以，这里也想多说几句，我个人觉得，好的C/C++程序员，应该做到自己的程序都能优雅退出，这是最基本的要求。这里所谓的优雅退出，指的是退出的时候把该释放的资源释放掉，且保证程序退出时不挂。

参考文献：http://code.google.com/p/google-perftools/wiki/GooglePerformanceTools