

# 云风的 BLOG

思绪来得快去得也快，偶尔会在这里停留

[« skynet 服务启动优化](#) | [返回首页](#) | [Ejoy2D 开源](#) »

## skynet lua 服务的内存管理优化

前两天一直困扰我的问题是并发启动 lua state 比串行启动它们要慢的多。而启动 lua state 的操作相互是完全无关的，没有任何应用层的锁。原本我以为多核同时做这些事情，即使不比单核快，也不至于慢一倍吧？

昨天有同学在 google talk 上和我讨论这个问题，说要不要考虑下是内存资源方面的问题。比如，在大量线程同时申请并读写大量内存时，有可能引起操作系统在映射物理内存到虚拟地址空间这个操作上出现性能问题。

虽然最后确认不是这个原因，但这启发我可能内存分配器在多核上工作并不顺畅。

我们使用 skynet 的项目利用 jemalloc 替换了默认 glibc 的分配器。tcmalloc 也应该是一个好的选择。但无论是哪个，都有多线程锁的问题。而 lua 可以自定义内存管理器，我们在 lua 服务启动时，若预分配 1M 内存，那么在这 1M 内存内的内存管理就完全没有线程安全的顾虑了，理论上这种定制的内存管理器会比一切通用管理器表现更好。

昨天我花了 3 个小时实现了一个简单的 lua 内存管理器，提交到 github 上。默认是关闭的，有兴趣使用的同学可以在 service\_lua.c 里把

```
#define PREALLOCMEM (1024 * 1024)
```

的注释去掉。这样就可以启用为 skynet 专配的 lua 内存分配器了。

实测这个分配器会比 glibc 的版本好 40% 以上，比 jemalloc 好 5% 左右。它虽然依然不能解释多核并发初始化 lua state 反而更慢的问题，但本身还是对性能有不少改进的。

云风 提交于 December 20, 2013 06:04 PM | [固定链接](#)

## COMMENTS

很好的支术

Posted by: [青岛瑛莎舍](#) | (8) [December 28, 2013 01:35 PM](#)

云风大哥，一直关注着你，看你的博客总能够收获许多无论是技术还是思想上的启发，真庆幸人生中有这么好的长辈给我指路，我的梦想是有一天也能写出高效的游戏引擎，目前正在努力，希望将来有一天能追随你的脚步，云风大哥，你要等着我呀！

Posted by: [晓锋](#) | (7) [December 27, 2013 11:23 PM](#)

大神啊,你走错道啊.想当前写出好用高效的2D引擎,很多公司争相使用.现在赶上了移动互联大潮,一大批公司有了好用的2D引擎的迫切需求,看看现在cocos2d-x搞得风生水起,无数千万级大作都在其上建立,手游行业典型的人傻钱多,小小的游戏让公司赚得钵满瓢满,错过这村没拿店了.这本是你最拿手的东,搞个有技术有内涵有想法的2D引擎那轮得到cocos2d.可现在你却在做着上游应用东,无法造福广大人民,想来搞出的游戏也不太会惊天地泣鬼神,最终沦为昙花一现.望大神多思考未来规划,造福我等小民.

Posted by: [guest](#) | (6) [December 27, 2013 08:28 PM](#)

气质好！程序员专用主页[yaodot.com](#)，代码日记本，搜索快速切换，IT网站导航，程序员必备！！！！

Posted by: [程序员主页](#) | (5) [December 27, 2013 12:24 AM](#)

tcmalloc对于小内存(<32KiB)有thread cache，多次分配小内存可以做到无锁的

Posted by: [mine260309](#) | (4) [December 23, 2013 02:35 PM](#)

真是简洁的好办法,nice

Posted by: coollen.mmx | (3) [December 21, 2013 11:34 AM](#)

这种在线性能问题，用systemtap不是能够很容易就查出么？

Posted by: heeroz | (2) [December 20, 2013 10:55 PM](#)

实际上在java技术中也有类似的技术，同样是位了解决多线程内存分配的同步问题--TLAB(Thread local allocation buffers)

Posted by: lee | (1) [December 20, 2013 10:21 PM](#)

## POST A COMMENT

非这个主题相关的留言请到: [留言本](#)

名字:

Email 地址:

为了验证您是人类，请将六加一的结果（阿拉伯数字七）填写在下面:

URL:

☐ 记住我的信息？

留言:

（不欢迎在留言中粘贴程序代码）