

编程语言

即时编译 (JIT)

虚拟机

动态语言

✎ 修改

## 在支持JIT的动态语言虚拟机中，是否有必要用局部变量缓存字段访问？✎ 修改

在动态语言如Lua中，对于下面的代码： local sum = 0 for i = 1, 1000000000 do sum = sum + s...显示全部

关注问题

✎ 写回答

+ 邀请回答

💬 6 条评论

🔗 分享

★ 邀请回答

🚩 举报

...

查看全部 7 个回答

 Felis sapiens ⭐

函数式编程、编程语言、编程 话题的优秀回答者

3 人赞同了该回答

动态OO语言的JIT做属性访问时都不会直接查Hash表。我查到的LuaJIT的优化方式如下：

Hash slot specialization: Hash table lookup for constant keys is specialized to the predicted hash slot. This avoids a loop to follow the hash chain. Pseudocode:

```
HREFK: if (hash[17].key != key) goto exit
HLOAD: x = hash[17].value
-or-
HSTORE: hash[17].value = x
```

**HREFK is shared by multiple HLOADs/HSTOREs and may be hoisted independently.**  
The verification of the prediction (HREFK) is moved out of the dependency chain by a super-scalar CPU. This makes hash lookup as cheap as array lookup with minimal complexity.

It also avoids all the complications (cache invalidation, ordering constraints, shape mismatches) associated with hidden classes (V8) or shape inference/property caching (TraceMonkey).

摘自lua-users.org/lists/lua...（网页快照）。这里17是指这个constant key由编译器预先算出的hash值。普通实现是现计算key的hash值然后访问，现在预先计算，加一步判断（判断失败的情形，如整个表被resize等等，则goto exit）；然后这步判断可以被从循环中提出来。这样一来在循环中查Hash表的开销就没有了，表指针直接加上这个槽的offset就能访问到。常数时间。优化动态属性访问还有其它方法，就是他提到的hidden class和property caching，感兴趣的可以去找V8/PyPy/TraceMonkey等对应的博客文章，我就不一一列举了。

编辑于 2015-03-29

▲ 赞同 3

▼

💬 13 条评论

🔗 分享

★ 收藏

❤ 感谢

...

更多回答

 rainoftime ⭐

编程语言、编程 话题的优秀回答者

15 人赞同了该回答

关注者

82

被浏览

6,090

他们也关注了该问题



关于作者



Felis sapiens

⭐ 函数式编程、编程语言、编程 话题的优秀回答者

👤 电影旅行敲代码、Antokha Yuuki、暮无井见铃也关注了她

回答

624

文章

40

关注者

14,871

已关注

发私信

相关问题

HotSpot VM JIT的编译产出，理论上能否被复用？ 4 个回答

JavaScript引擎、虚拟机、运行时环境是一回事儿吗？ 7 个回答

如何通俗易懂地介绍「即时编译」（JIT），它的优点和缺点是什么？ 10 个回答

如果没有PGO，JIT 编译相比AOT 编译有哪些优势？ 4 个回答

Go 语言的设计为什么没有虚拟机？ 28 个回答

一、动态语言的快速属性访问

1. 问题引入

举个栗子，看如下JS代码

```
function Point(x, y) {
  this.x = x;
  this.y = y;
}
var point = new Point(2, 3);
point.x = 4;
point.y = 5;
```

展开阅读全文

赞同 15

11 条评论

分享

收藏

感谢

...

 **冯东**  编程、Linux 话题的优秀回答者

5 人赞同了该回答

现实是，「如此机械的手工操作」还真不是能由「虚拟机代劳的」。你的手工优化有 Lua VM 不知道的创造性知识：

- function string.byte() 不会修改 string
- 展开阅读全文

赞同 5

1 条评论

分享

收藏

感谢

...

查看全部 7 个回答

相关推荐



**森懂物理学：理解世界的极简指南**

共 31 节课

试听



**自己动手写 Java 虚拟机**

张秀宏

5 人读过

阅读

知乎

**超级会员**

戳此进入>>>

3000+ 场好课随意听

1000+ 本好书任意读

广告

刘看山 · 知乎指南 · 知乎协议 · 隐私政策

应用 · 工作 · 申请开通知乎机构号

侵权举报 · 网上有害信息举报专区

违法和不良信息举报：010-82716601

儿童色情信息举报专区

电信与服务业务经营许可证

网络文化经营许可证

联系我们 © 2018 知乎