

《C 0x漫谈》系列之：右值引用 PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/264/2021\\_2022\\_\\_E3\\_80\\_8AC\\_\\_0x\\_E6\\_BC\\_c97\\_264475.htm](https://www.100test.com/kao_ti2020/264/2021_2022__E3_80_8AC__0x_E6_BC_c97_264475.htm) 右值引用（及其支持的Move语

意和完美转发）是C 0x将要加入的最重大语言特性之一，这点从该特性的提案在C - State of the Evolution列表上高居榜首也可以看得出来。从实践角度讲，它能够完美解决C 中长久以来为人所诟病的临时对象效率问题。从语言本身讲，它健全了C 中的引用类型在左值右值方面的缺陷。从库设计者的角度讲，它给库设计者又带来了一把利器。从库使用者的角度讲，不动一兵一卒便可以获得“免费的”效率提升... Move语意 返回值效率问题返回值优化（(N)RVO）mojo设

施workaround问题定义Move语意语言支持 大猴子Howard Hinnant写了一篇挺棒的tutorial（a.k.a. 提案N2027），此外最初的关于rvalue-reference的若干篇提案的可读性也相当强。因此要想了解rvalue-reference的话，或者去看C 标准委员会网站上的系列提案（见文章末尾的参考文献）。或者阅读本文。

源起《大史记》总看过吧？故事，素介个样子滴...一天，小嗖风风的吹着，在一个伸手不见黑夜的五指... 我用const引用来接受参数，却把临时变量一并吞掉了。我用非const引用来接受参数，却把const左值落下了。于是乎，我就在标准的每个角落寻找解决方案，我靠！我被8.5.3打败了！... 设想这样一段代码（既然大同小异，就直接从Andrei那篇著名的文章里面拿来了）：

```
std::vector v = readFile(). readFile()的定义是这样的：  
std::vector readFile() { std::vector retv. ... // fill retv return retv. }
```

这段代码低效的地方在于那个返回的临时对象。一整

个vector得被拷贝一遍，仅仅是为了传递其中的一组int，当v被构造完毕之后，这个临时对象便烟消云散。这完全是公然的浪费！更糟糕的是，原则上讲，这里有两份浪费。一，retv（retv在readFile()结束之后便烟消云散）。二，返回的临时对象（返回的临时变量在v拷贝构造完毕之后也随即香消玉殒）。不过呢，对于上面的简单代码来说，大部分编译器都已经能够做到优化掉这两个对象，直接把那个retv创建到接受返回值的对象，即v中去。实际上，临时对象的效率问题一直是C中的一个被广为诟病的问题。这个问题是如此的著名，以至于标准不惜牺牲原本简洁的拷贝语意，在标准的12.8节悍然下诏允许优化掉在函数返回过程中产生的拷贝（即便那个拷贝构造函数有副作用也在所不惜！）。这就是所谓的“Copy Elision”。为什么(N)RVO((Named) Return Value Optimization)几乎形同虚设 还是按照Andrei的说法，只要readFile()改成这样：

```
... readFile() { if(/* err condition */)
return std::vector(). if(/* yet another err condition */) return
std::vector(1, 0). std::vector retv. ... // fill retv return retv. }
```

出现这种情况，编译器一般都会乖乖放弃优化。但对编译器来说这还不是最郁闷的一种情况，最郁闷的是：

```
std::vector v. v =
readFile(). // assignment, not copy construction
```

这下由拷贝构造，变成了拷贝赋值。眼睛一眨，老母鸡变鸭。编译器只能缴械投降。因为标准只允许在拷贝构造的情况下进行(N)RVO。为什么库方案也不是生意经 C 鬼才Andrei Alexandrescu以对C标准的深度挖掘和利用著名，早在03年的时候（当时所谓的临时变量效率问题已经在新闻组上闹了好一阵子了，相关的语言级别的解决方案也已经在02年9月份粉墨登场）就在现有

标准（C 98）下硬是折腾出了一个能100%解决问题的方案来。Andrei把这个框架叫做mojo，就像一层爽身粉一样，把它往现有类上面一洒，嘿嘿...猜怎么着，不，不是“痱子去无踪”：P，是该类型的临时对象效率问题就迎刃而解了！

100Test 下载频道开通，各类考试题目直接下载。详细请访问  
[www.100test.com](http://www.100test.com)