



将CPU PMU引入Java世界

工作历史纪要及现状—长仁

开篇闲话，PMU的意义

- **PMU**是什么。
- 在“性能”上下文下，利用**PMU**是基本功。
- **X86 ias**下实现厂家**Intel**，**AMD**，**VIA**都没有公布hdl。甚至对公司内。
- **PMU**探究微架构状态的手段，众多学术界微架构模拟软件只是“猜测”，是做它用的。

开篇闲话，软件开发者步入硬件世界需要自毁三观

- 拿PMU举例，毁世界观：
 - 软件实现1k个事件监控需要多少代价？
 - 硬件实现1k个事件监控需要多少代价？
 - 一个通俗的例子



认识PMU，只讲手册里没讲过的

- 存在于众多CPU架构中，狗剩都有好多个
- PMU的大招
 - 计数
 - 中断
- 可以监控的事件繁多？多少都是少！
- 常规用途，有可替代方法
 - 找热点
 - 找瓶颈
- 重要用途，真正体现它的价值
 - “显微镜”
- Geek用途，使用的极致
 - “表哥”
 - “Holter”
 -

开始说Java相关

- 2年前，如果用以上工具诊断Java应用，你会很失望。
 - 能看到jvm实现部分及native部分。
 - 不能看到你更关心的java程序实现部分。
- 对于jvm优化现有工具够了，对于java开发同学，相当于没有工具可以利用PMU。
- 无论java开发需要不需要，提供除了timer based或instrumentation以外的更准确，全面，开销小的选择都是必要的。

无法入选的选手

- **AMD CodeAnalyst**

- 我们阿里巴巴本采用**AMD**处理器
- **CodeAnalyst**能对**Intel CPU**提供很好的支持也许要等到**vtune**对**AMD CPU**提供很好的支持之后☺

第一位出局的选手 oprofile-2年前

- 优缺点
 - 各家CPU及不同微架构不同，亚历山大
 - 架构支持有限，支持不够
 - 开源的小众也有观众。需要厂家支持才能活，比如狗剩
 - 功能架构够了
 - 社区小，不好借力
- 出局过程
 - 沦为timer based工具
 - 即使有Jit api，以及openjdk vm中的支持，然后只能黯然退场。

悲情的一号种子vtune-2年前

- Jvm中曾经存在的vtune开头的预留接口
 - 尝试对接vtune manipulation api，实现成功。
 - 随后的日子非常美好，直到有一天
 - 新版jvm vtune相关接口被干掉。
 - Vtune manipulation api真的就不支持了？
 - 它地，闭源的家伙大大地，我地，无能为力
- 和intel vtune开发团队负责人开会，强调需求，获知最新版本vtune隐藏着一个实验参数，验证可以工作
 - 随后在一起生活的日子里，岁月不断放大对方的缺点
 - Vtune和实验功能的bug们
 - 功能限制，后娘养的，系统级监控，attach都不支持

新星 perf-1年前

- 优缺点
 - Kernel社区里好热
 - Intel人员参与，大义工，支持不错，并且可以部分“强”
 - Kernel内事件，trace功能囊括在一起，大有舍我其谁之势
 - 年轻，bug太多了
 - 对jit考虑机制还没有，必须从无到有开大手术

perf jit接口及实现-现在

- 目前Taobao kernel perf jit支持以及和taobao jvm配合
 - kernel patch
 - 共享内存分享symbol，作为接口，以后可能提供api
 - 快速分享，无锁设计
 - 监控java jit结果，也可监控其他jit，例如lua jit等
 - Jvm patch
 - 利用oprofile jit支持的尸体
 - 小修，避免撑死perf
 - Jvmti agent
 - Taobao kernel和JVM支持。Kernel部分我已经交给kernel组承钢同学维护，jvm及jvmti部分，交给成滔同学维护

PMU对Java世界的作用及后续

- 常规用途用Java开发有用，所以perf，vtune都是好工具
- 重要用途对JVM优化有用，例如intrinsic实现的优化。
- System wide profiling我们也会考虑支持Java JIT支持

没有广告的分享不是好广告，谢谢

- 专用计算组欢迎内部转岗及社招推荐
 - JVM优化
 - 性能优化
 - 功能增强
 - Bug修复
 - 专用计算
 - 针对特定计算密集型应用
 - 针对CPU微架构的算法及实现优化
 - GPU应用及算法及实现优化
 - 目前进行中项目
 - 图像搜索算法实现优化
 - 机器学习，逻辑回归实现算法优化
 - MPI实现优化
 - 和Java语言、JVM无关