存储技术基础论文阅读报告

用户态文件系统FUSE的综合性能测试

To FUSE or Not to FUSE: Performance of User-Space File Systems

阅读人：计43 黄家晖

用户态文件系统由于其实现、调试方便的特点，当今不仅被大量用作系统原型实现，也常被用于生产环境文件系统（例如GFS, GlusterFS）的实现，但目前很少有文献对用户态文件系统的性能损失进行过具体测量。该论文则填补了这个缺漏，在单机下使用不同负载测试了FUSE这一用户态文件系统实现技术与内核态文件系统的性能对比，并针对FUSE近日做出的各类优化进行了有效性的分析和测量。该论文发表在FAST17上，第一作者是Stony Brown大学FSL实验室的研究生B. K. R. Vangoor。

作者所测试的目标文件系统为自己开发的StackFS，该文件系统将FUSE请求直接转换成针对内核态Ext4文件系统的调用。在对比时，作者构造了42种负载组合以及3种真实服务器文件系统负载为测试集合，在单核/多核、HDD/SSD设备上分别测量了FUSE相比于Ext4内核文件系统吞吐率和CPU利用率差距。由于在最新版本中实现了用户-内核零拷贝、多线程以及Write-back Cache等优化，作者还对优化前后的用户态文件系统实现效率进行了比较。

结果表明，FUSE带来的性能损失随着负载不同而不同。最差情况下吞吐率较内核态FS损失达到83.1%，而最好情况下吞吐率反而增加了6.3%；CPU的利用率较内核态FS增加了0.13%到31.2%不等；进行过优化的FUSE相比于未优化版本吞吐率平均提升45%。FUSE在进行文件元数据操作的时候带来的吞吐率下降最明显，但总体来说经过了优化的FUSE与原生Ext4相比平均吞吐率差距不过5%。

本问的亮眼之处就在于，测量场景非常全面，并几乎对每种情况给出了理论、客观的分析，对于希望快速了解结果的读者也贴心地提供了Overview部分，且是为数不多几篇针对FUSE的优化技术进行对比测量的文章之一。然而，本问仅仅分析了单机系统上的StackFS的相关开销，而目前生产环境中的用户态文件系统则多基于网络，有时网络方面的开销远远大于单机上的内核-用户态转换开销、才是主要瓶颈，导致本工作的测量结果可用性不是很广，实际评价也应综合考虑性能提升与实现复杂度的制约关系。另外，本测量基于Linux，对于Windows服务器用户态文件系统的测量也可以作为下一步工作之一。

在读完这篇文章之后，我了解到了进行测量型科研工作对于结果以及分析严谨性极高的要求，同时也学习到在评价一项技术或其优化的时候一定要基于其具体应用场景、从多方面进行评价，不应该一概而论。不单是FUSE，所有技术都有着进一步探索和提升的空间待我们去发现。