2011年12月23日 淘宝核心系统数据库组 褚霸做的PPT

MySQL和IO

blog.yufeng.info

不仅仅是讲MySQL还讲了涉及到的方方面面。硬件、OS、存储~

##############################################

提纲：

硬件

操作系统

innoDB引擎

MySQL

Flash存储设备的选择

**硬件：**

CPU和内存

NUMA架构

缓存践踏

大内存管理开销：大页

RAID卡：

内置缓存：WB还是WT

读写缓存比例

预读

BBU放电问题

PCIe Flash卡和SSD

Flash不担心并发IO问题。NCQ:

IOPS多剩问题

中断亲缘和平衡问题：对应用的影响

供应商和产品特点：

fusionio

virident

华为等国内厂商

Flash存储设备的选择：成本与性能

用途分类：读密集：SSD+RAID

写密集：FIO(+flashcache)

容量密集：FIO+ flashcache

未来的展望：设备的层次结构无论如何都会存在，cache不会消亡，更块的设备会很快出现，系统和数据机构有很大改变，传统数据库需要跟进。

**操作系统**：

操作系统部应该再是黑盒子，追新无罪。

**内核：**

RHEL 5U4还是6U1

超大2M页面

页面回写per设备

页面回收split lru

**文件系统：**

ext3和xfs

数据根据底层设备只能对齐，对ssd友好

单个文件和并行dio

更快的文件追加速度

更少的锁冲突

mount选项

-nobarrier

-data=ordered,writeback

预读真的有用么？

**页面缓存：**

vm.swappiness=0

vm.dirty\*

vm.pagecache

posix\_fadvise清理buffered io引起的垃圾页面

sync\_file\_range强制页面回写

fsync天花板：如何计算、测量

**块设备：**

队列调度算法

-deadline或者noop

-cfq害人不浅，顺序变离散

-请求队列长度和latency

flashcache

担心uncached IO

以2M大小的set为单位进行脏页回写

dev.flashcache.skip\_seq\_thresh\_kb

软raid

开销不大，整合设备

**InnoDB存储引擎**

又一个操作系统，各种微调

**引擎如何工作**

引擎核心部件

库、表、行、列、事物概念，各种统计实现

核心数据结构BTREE

内存管理，buffer和page

逻辑文件空间管理，IO线程池管理读写请求，同步和异步

cache淘汰，回写

和操作系统职能冲突

谁更明白用户的需求

如何调优

BP是核心

**大话MySQL**

MySQL架构

网络层太老，HS这样的新架构出现

SQL层和存储层分离，层间数据交换开销大

锁开销太大

对NUMA架构不友好

网卡

网络丢包重传thin tcp