本文来自阿里巴巴，他们分析了阿里云的SSD存储系统和它的各种故障，并总结了一些经验与教训。

## 研究背景

存储系统对当今互联网世界的重要性，越来越多的企业开始使用大量的ssd存储系统，包括大家熟悉的腾讯云，亚马逊云，阿里云。关于SSD也有非常多的相关研究，但更多关注的是SSD的本身性能，本文则选择了从更宏观的角度研究SSD相关问题

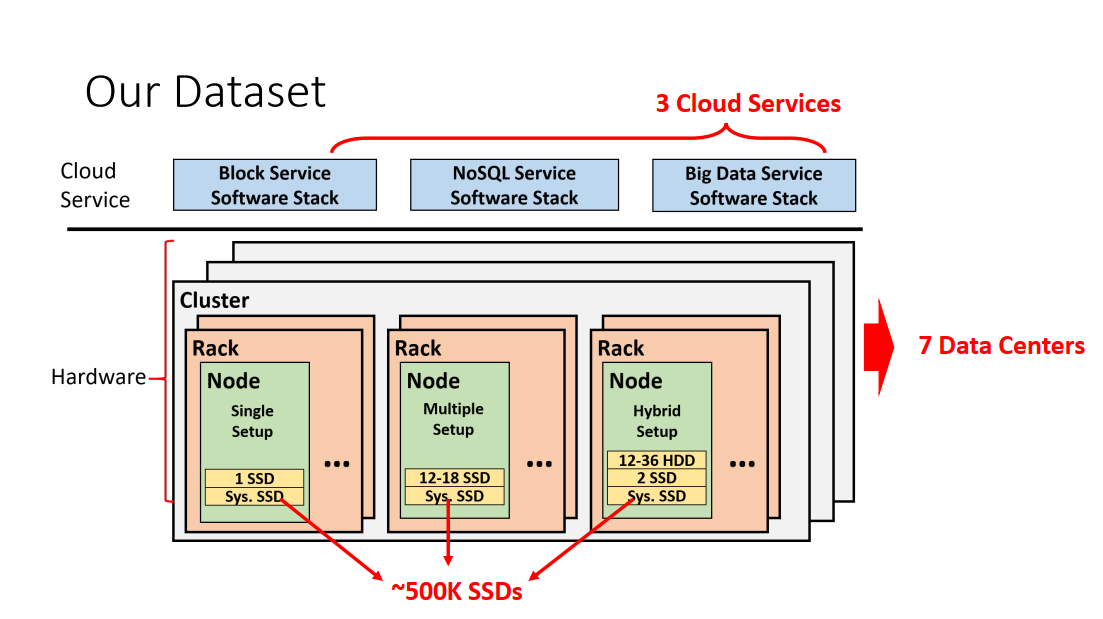
这里有一个比较高层次的概括，下面这一部分是各种各样的存储节点，主要有三种配置，

最左边有一个额外的ssd的节点，是主要的计算节点，有非常强力的CPU

右边有12到36HD盘和两个SSD盘，这是HDD存储节点，这两个SSD盘主要是用于缓存

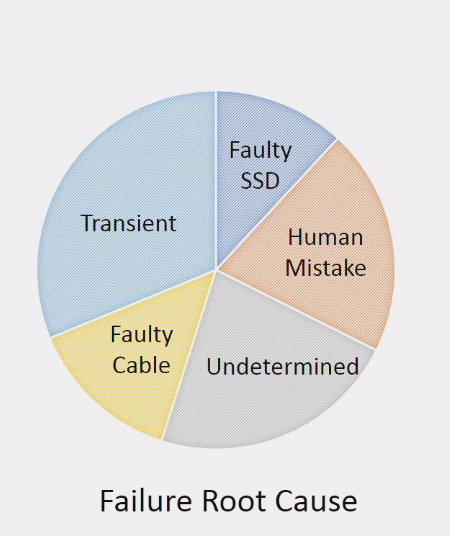
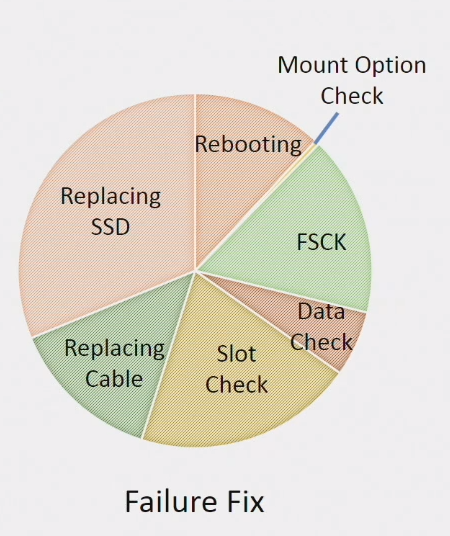
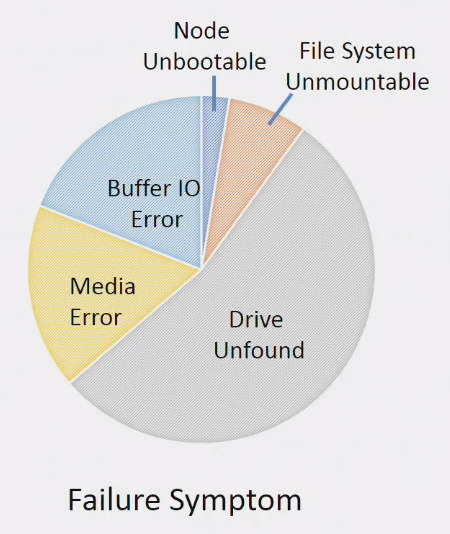
中间的有12到18个SSD盘的SSD存储节点

每一个机架上大概有16到48个同类型的节点，这些所有节点组成集群，为上层各种各样的服务提供支持，这里的服务主要是讲了块存储服务，NoSQL，大数据。在这样一个基础上他们收集了大量的日志文件，通过这些日志文件来分析出碰到了哪些硬件软件的问题。

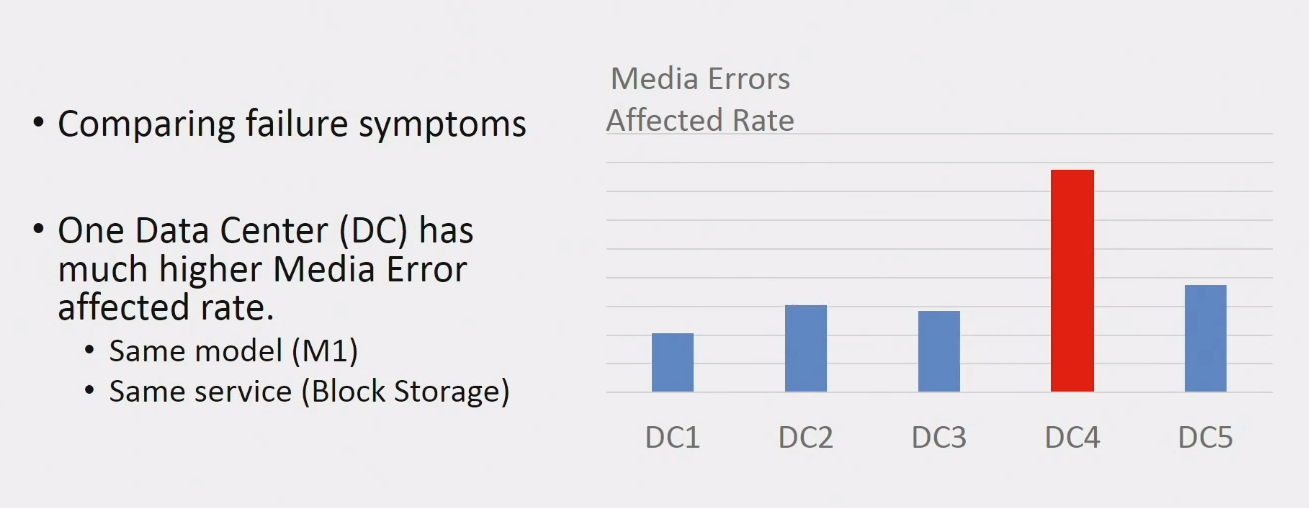


## 问题概述：

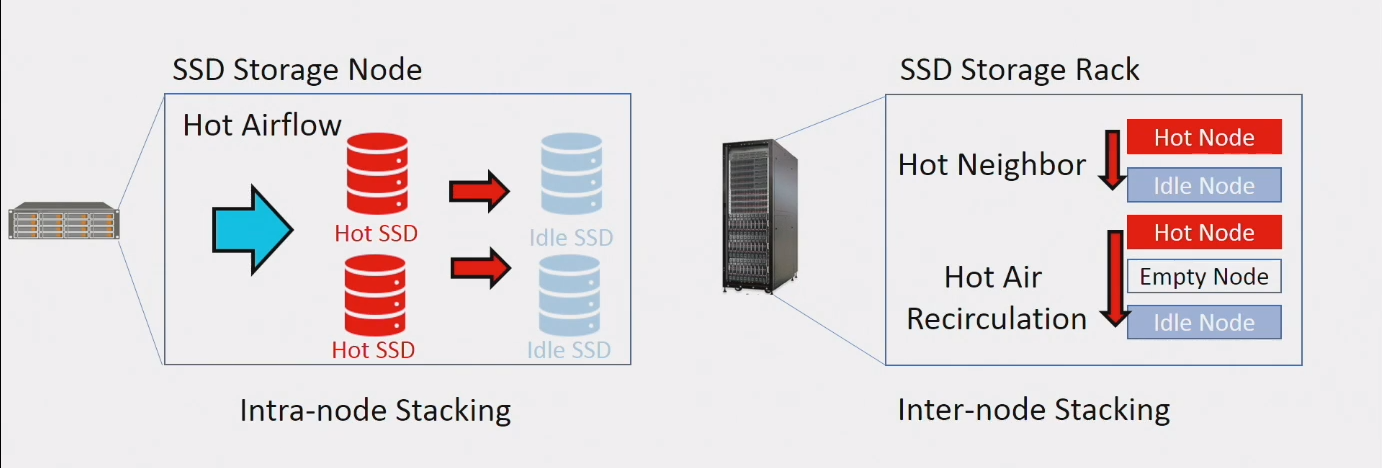
通过实地考察，日志分析，以及实验等一些方法。三年内总共收集了十五万的错误案例，RSAR错误所占比例大概百分之六，把这一万个错误，进一步拆分，主要会有这五种症状，节点无法启动，文件系统无法挂载，设备无法找寻，缓存io错误，以及Media Error。这些错误又会有相应的修复方法，和对应的原因。



## 发现问题与解决方法

在统计了所有的SSD故障之后，研究人员发现了一个很奇怪的现象，在使用相同型号的盘和进行相同的服务的时候，在DC4数据中心，Media Error的故障率特别高。

经过实地考察之后，发现DC4相比于其他数据中心，SSD存储节点比较多，导致高速运转的发热SSD，在面对冷风的时候，会把这部分热量吹给闲置的SSD。此外，在DC4有大量的计算节点和存储节点混合布置，导致在DC4里面很多SSD会处于一个被动加热的情况，导致温度升高，并进一步导致数据出现大量错误。



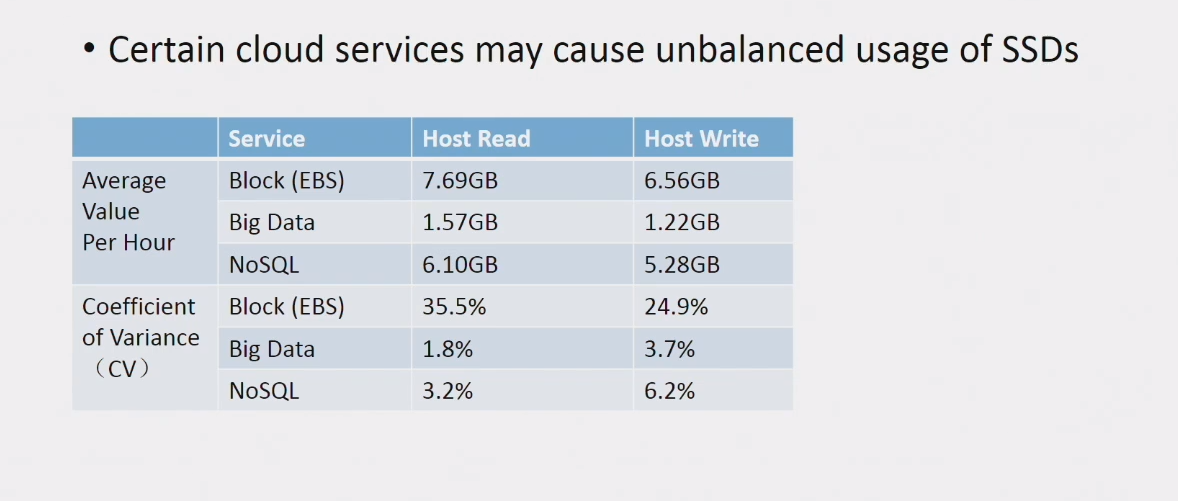
进一步做了一个小型实验节点分析发现，即便只是加热128小时，即便只是正常工作温度55度，处于这种被动加热的SSD仍然会导致多达百分之57的错误

为了解决此问题提出了两种方法，一种是基于软件对SSD例行扫描，大概每4小时一次。

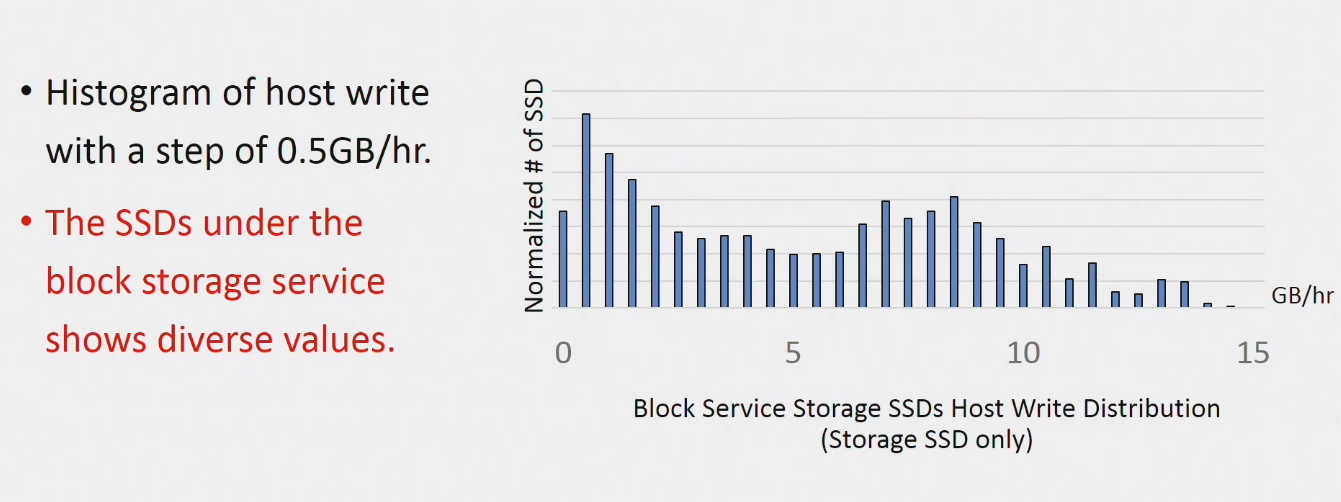
这利用了SSD的Read Fresh的特性，当在读取SSD内容的时候，如果SSD发现该部分内容有一定的缺陷，它就会进行重新的写入。为了减少这种由于被动加热导致的数据丢失，可以定时的对全盘数据进行一次扫描，但这样的代价是扫描可能会引入更多的读取干扰错误。

第二个是建议硬件厂商把被动加热的监控直接整合到FTL里面去，可以做到自动检验和纠错的话，软件层不必做任何改动。

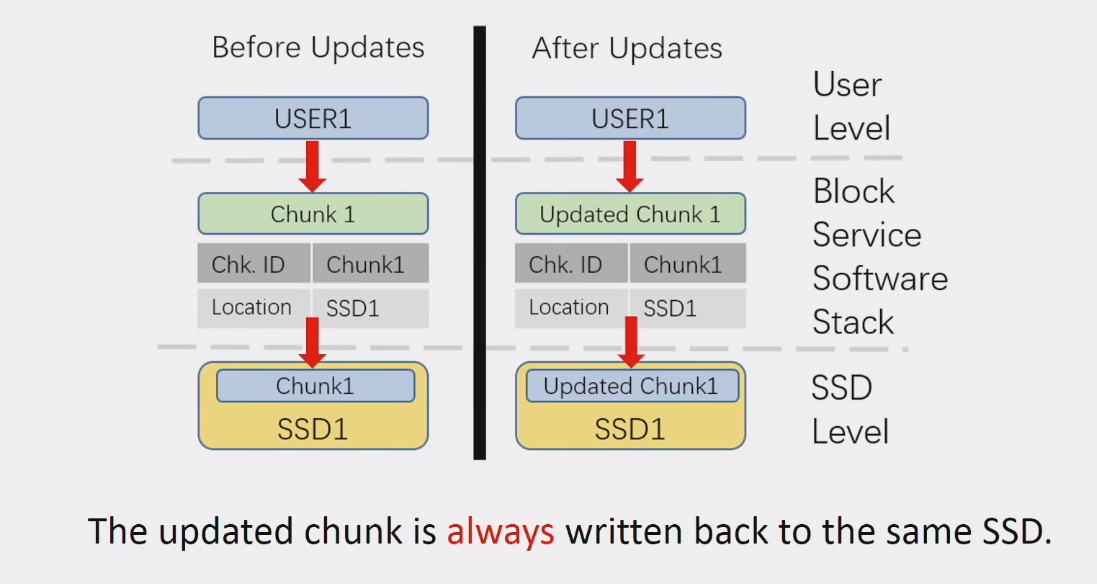
而在软件层上，对SSD影响最大的是对其读写，三种服务的平均读写差异并不大， 但进一步分析服务读写的变异系数（CV）发现对于块存储服务CV值非常高，这往往代表这个服务下的SSD处于非常不平衡的使用。



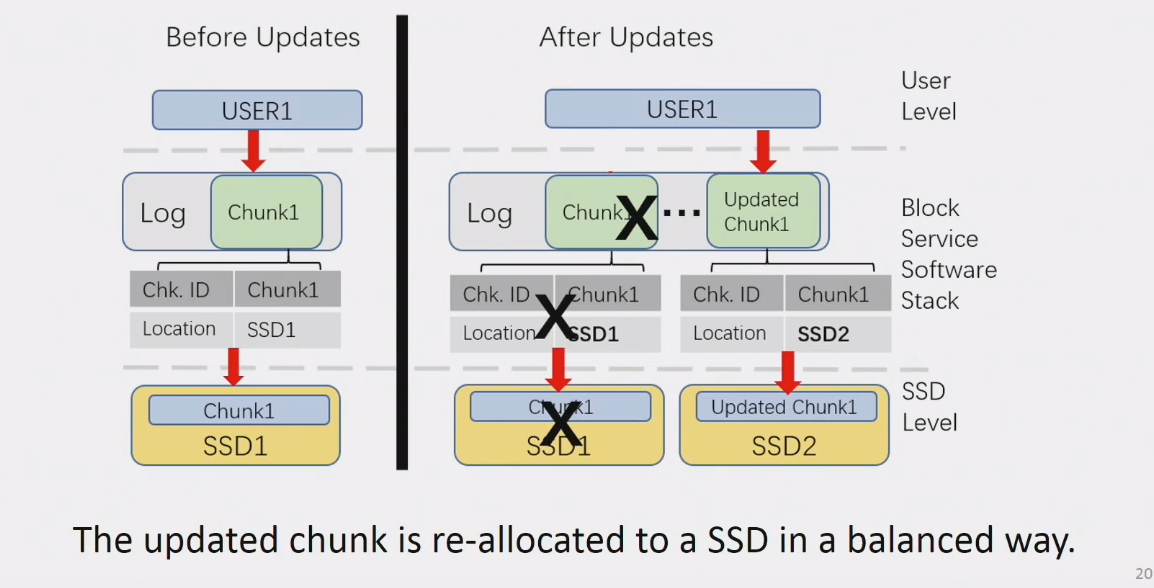
进一步的对块存储SSD的写入量进行了分析，发现分布得非常不均衡。



通过和块存储程序员沟通，在之前的块存储服务中，用户选择存储一部分数据的时候，这个数据会被映射成一个chunk，这个chunk会和一个SSD的位置进行绑定，这个chunk就会被写入这块SSD，这一块区域就归这个用户所有。当用户需要修改数据的时候，就会把这个chunk进行一个更新，这个更新后的chunk就会根据之前的mapping table回到相应的SSD上把内容进行改动，这样实现非常快捷方便。但是坏处是，有的用户写入量会远远大于其他的用户，就会导致某一些SSD相比于其他的SSD使用量会变得尤其的大，这些SSD就容易更早的出现问题。



为了解决这个问题，通过添加一个共享的日志来优化块服务的软件堆栈，当用户新增加内容的时候，新的chunk就会被加入到这个log的尾部形成一个新的mapping table，把数据写入到当时最为合适的SSD上。这样好处在于每次写入数据的时候都是选择一个相对平衡的策略，而不是写入同一个SSD。由此以来就解决了软件上的问题。



运维的问题其实更多的是人为的问题。

首先来看一下Drive unfound failures，是所有碰到的5种RASR 错误里面最频繁的一种情况，主要的维修方式是以下四种，重启，检查接口，更换线缆，更换SSD。因为错误日志只会告诉发生了什么错误，但是并不会告诉你错误的原因，所以维修人员就只能按照一定的顺序从软件到硬件逐一排查，耗费时间。

于是如果能够通过一些方法判断错误的原因，就可以节省很多的工作时间。于是他们想到了利用CRC错误，CRC就是盘在传输的过程中需要做的一次校验，如果频繁的出现CRC错误的话，说明盘的链接很可能是有问题的。所以这里列举了4种维修方式，以及每种维修方式里面出现CRC错误的多少，可以看到如果大量的出现CRC错误的话，盘出现Drive unfound 错误的话，很有可能是你的线有问题。发现这种情况，日志里面有很多CRC 错误，就可以叫维修人员带着一根线去换就行，而不用重新换块盘。

在各类错误里面，还有一个占很大比例的错误是认为错误，由于盘具有不同的功能，如系统盘，journaling盘，storage盘，由于这些盘在硬件接口上之前都是SATA，有时候就会出现把系统盘插到其他功能的盘上，就会导致出现一些问题。比如节点无法启动等等。

随着现在技术的发展，有了越来越多的硬件接口，比如M.2 ,U.2，PCIE，于是就采用了OIOP的一个可行策略，把软件的功能和硬件的接口进行绑定。比如系统盘就用M.2的SSD，如果是缓存盘这些性能为重的盘就用PCIE的SSD，如果是传统的存储盘就直接用SATA SSD，这样就可以有效减小错误的发生。

## 总结与不足

文章太过偏于工程实践，在拥有一个如此大规模的ssd存储系统的情况下，完全可以挖掘出更多的东西，而本文只是简单的发现问题，解决问题，没有更深度的研究问题的成因和进行更多的研究。