存储

次世代数据存储思维与技术

冬瓜哥 著

清華大學出版社

存储传

次世代数据存储思维与技术

冬瓜哥 著

清華大學出版社
北京

内容简介

本书为《大话存储终极版》出版以来最新积累的大量高质量技术热点内容。

全书分为：灵活的数据布局、应用感知及可视化存储智能、存储类芯片、储海钩沉、集群和多控制器、传统存储系统、新兴存储系统、大话光存储系统、体系结构、I/O协议栈及性能分析、存储软件、固态存储等，其中每章又有多个小节。每一个小节都是一个独立的课题。本书秉承作者一贯的写作风格，完全从读者角度来创作本书，语言优美深刻，包罗万象。另外，不仅阐释了存储技术，而且同时也加入了计算机系统技术和网格技术的一些解读，使读者大开眼界，茅塞顿开，激发读者的阅读兴趣。

本书适合存储领域所有从业人员阅读研习，同时可以作为《大话存储终极版》的读者的延伸高新资源。

图书在版编目(CIP)数据

大话存储后传：次世代数据存储思维与技术 / 冬瓜哥著. — 北京：清华大学出版社，2017

ISBN 978-7-302-46492-1

Ⅰ. ①大… Ⅱ. ①冬… Ⅲ. ①数据存贮—研究 Ⅳ. ①TP333

中国版本图书馆 CIP 数据核字(2017)第 025580 号

责任编辑： 栾大成
封面设计： 杨玉芳
版式设计： 方加青
责任校对： 胡伟民
责任印制： 沈 露

出版发行： 清华大学出版社
网 址：http://www.tup.com.cn，http://www.wqbook.com
地 址：北京清华大学学研大厦 A 座 邮 编：100084
社 总 机：010-62770175 邮 购：010-62786544
投稿与读者服务：010-62776969，c-service@tup.tsinghua.edu.cn
质 量 反 馈：010-62772015，zhiliang@tup.tsinghua.edu.cn
印 装 者： 北京鑫丰华彩印有限公司
经 销： 全国新华书店
开 本： 170mm×240mm **印 张：** 29.75 **字 数：** 745 千字
版 次： 2017 年 5 月第 1 版 **印 次：** 2017 年 5 月第 1 次印刷
印 数： 1～4000
定 价： 89.00元

产品编号：073341-01

序言

不用扬鞭自奋蹄！

廖恒　博士

是哪一年认识了张冬，我想不起来了。只记得是一次出差途中，在杭州一家小餐馆里一起进餐。见面寒暄后，他单刀直入问我：时钟具体是如何驱动电路的运行的？带着孩子般的渴望知识的眼神，透露着作者的气质，渗透在他的每一本著作中。此后每次会面都是擦肩而过，每次碰撞都留下一个不同的问题。我常常回味，驱动他努力追求的难道仅仅是好奇？年岁易过，技术领域更新极快，方才还潮流新宠的新课题，转眼已融化殆尽，抛入了记忆的废纸篓。热心收集的知识岂不成了占内存的老古董，有何用现实的价值？

拿到这本“重达”40M的电子版大部头，我着实被吓了一跳。这是一位孜孜不倦的探险者，用自己的眼睛去增长见识，用自己的心去思考实质，再用自己的笔去分享观感。内容翔实，让人耳目一新。好比为读者打开了私藏的博物馆，而由收藏主人亲自展示每一个藏品的精妙机关，再把当初苦心寻访藏品并终于纳入囊中的故事向你娓娓道来。其中扑面而来的喜悦，只有同道中人才能体会。

《大话存储后传》给出的是作者的答案。多年的追求探索，不仅仅是加深自身领悟，还为了和更多人分享和传承。工程师担负了造新物的使命，要看清这无比复杂的知识世界十分不易。冬瓜哥帮我们梳理了经纬全局，把知识的珠子串成了项链。

完成这一本，存储整个博物馆，就此剪彩全面开张，存储这档子事儿也许到此告

一段落。不知道那孩童般的好奇又会把张冬带到哪个新奇的世界。我期待他的下一本游记。

廖恒　博士

廖恒 博士，曾就读清华大学、美国普林斯顿大学。曾任PMC-Sierra公司Fellow，曾参与T10 SAS 标准制定工作，并担任存储部门总架构师，设计了SAS Expander、RAID控制器、HBA控制器等产品。

变化的IT，变化的存储

雷迎春 达沃时代CTO

CPU向两个方向发展：高性能和低成本。CPU的高性能使得对于大多数应用，只需要CPU的10%或30%的处理能力就可满足应用所需，为利用富裕的CPU处理能力，以VMware和Docker为代表的CPU虚拟化技术先后出现，帮助应用以不同的隔离形式并行运行，复用CPU资源。CPU的低成本使得计算机不再高大上，而是以各种形式，如手机、IoT设备等，充斥在人们的生活中，数据生产成本大幅下降，迎来大数据时代。

VMware的CPU虚拟化技术把一台物理服务器虚拟化为多台虚拟服务器，即虚拟机，从而允许在每一台虚拟机内运行独立的操作系统和应用。显然，只要在一台物理服务器上运行合适数目的虚拟机，CPU资源就能得到充分利用。不过，虚拟机在充分使用CPU资源的同时，对存储资源的使用也显著增加，而且以随机I/O为主。在虚拟化技术兴起之前，存储的主要产品形式是磁盘阵列，但是，磁盘天生就不适合支持随机I/O，所以，磁盘阵列很难适应虚拟化技术的飞速发展。更擅长随机I/O的闪存被用于改造以磁盘为中心的传统阵列，使阵列演变为混合闪存和磁盘的混合阵列，以及全闪存阵列。

相比于单个盒子的阵列，近年来兴起的分布式存储因较强的横向扩展能力而具有明显的优势。一般地，分布式存储由若干节点组成，每一个节点是一个中、小型存储服务器，它们通过分布式存储软件汇聚为一个大型的存储系统。在分布式存储中，新增一个节点，不仅增加整个系统的存储容量，同时，也提升整个系统的I/O性能。另一方面，当一个节点发生故障时，不会影响整个系统的正常运行，因为，故障节点的数据在其他存活节点上有冗余（副本），且存活节点能继续对外提供服务。由于体系结构上的优势，分布式存储不再有类似阵列存储的性能瓶颈和容量瓶颈。

（1）从消费级市场步入企业级市场的闪存是这一轮存储变革的关键因素。在以磁盘为中心的传统存储中，其硬件平台和软件系统是针对磁盘特别设计的，多年发展，积累至今。用闪存盘简单替换硬盘，会让传统存储的性能有所改善，但是，面向磁盘设计的存储软件并不能充分发挥闪存的性能特性，相反，由于闪存有不同于硬盘的物理特性，如有限的可擦写次数，而扩大闪存的缺陷，给系统带来隐患。因此，存储软件需要重构，从过去以磁盘为中心发展到现在的以闪存为中心。

（2）软件定义存储的兴起是这一轮存储变革的关键推手。过去，磁盘阵列有专门设计的硬件平台，而软件定义存储的理念是存储系统的软件化，不使用专门为存储业务特殊设计的硬件平台，而使用标准化硬件平台，如x86服务器。软件定义存储既允许存储厂商销售软硬一体的设备，又允许存储厂商直接销售存储软件给用户，用户自己选择硬件平台；存储的主要产品形式既可以是阵列，也可以是分布式存储，如超规模（Hyperscale）或超融合（Hyperconverged）。这里，超规模是应用软件运行在计算服务器节点上，存储软件运行在存储服务器节点上，存储和计算分离；超融合是一台服务器上同时运行应用软件和存储软件，存储和计算融合。软件定义存储带来最重要的特性是，存储软件和硬件平台解耦，允许分别升级。例如，如果发现存储性能与硬件平台有关，就升级硬件平台；如果存储性能与存储软件有关，或有最新发布的存储功能，只需要软件升级。简而言之，软件定义存储带来极大的灵活性和成本的降低。

（3）应用部署广泛采用虚拟机或容器技术，以及企业数据中心从第二平台向第三平台的演变左右着这一轮存储变革的进程。例如，正是虚拟机的大规模使用，磁盘型存储系统难以支持，才促进闪存型存储系统的发展，而企业数据中心从个人电脑、客户端/服务器和局域网/互联网为依托的第二平台转向以云计算、大数据、移动、社交为依托的第三平台，相应地，信息技术的价值从以计算、业务为中心转向以用户、数据为中心。伴随IT生态的变化，更适合第三平台的分布式存储的重要性日益凸显，市场份额显著增长，且软件定义存储的理念贯穿其中。特别地，在第三平台中，同一个共享存储池上需要运行各种应用，那么，分布式存储需要提供多种存储访问协议，如iSCSI、NFS/CIFS和S3等，且适应不同工作负载对存储资源的并行访问。

个人认为，在第三平台时代，存储技术会随着无处不在的数据和对数据处理的友好支持而百花齐放，存储也因为结合丰富的数据服务和数据管理功能，而模糊与应用之间的关系，出现应用驱动的存储或应用定义的存储。例如，Web对象存储以Web服务的形式对外提供存储功能，VSAN为VMware vSphere提供以VM为中心的存储，HDFS为Hadoop计算框架提供专属存储。

正在发生的存储变革是多种力量综合博弈的市场结果。几乎任何一项被主流市场接受的存储技术，无论硬件还是软件，都有它的前世今生，只有了解过去，才能认识当下，《大话存储后传》正是这样一本承上启下的书，它可以帮助存储开发者和存储使用者深刻理解存储技术的点点滴滴和变化的过程。此外，该书的内容质量堪称顶级，全书内容均为冬瓜哥亲手炮制，处处体现了作者清晰、深刻的思维，描述技术入木三分，穷根究底，来龙去脉一目了然。能达到这种境界，需要多年的异于常人的学习和积累，更重要的是付出比常人更多的毅力和思考过程。我相信若非冬瓜哥的兴趣

和情怀驱动，是很难做到这个程度的。

纵观当即社会，在互联网影响下，一股浮躁的风气弥漫着各个领域。在这个大的时代背景之下，能够潜心研究，耐得住寂寞笔耕不辍分享给大众，坚守情怀，实则是难能可贵之事，社会需要更多的像冬瓜哥这样的人。

雷迎春　博士　达沃时代CTO

前言

眨眼间，距离《大话存储》一书出版已经8年了。在这8年间，冬瓜哥也一直在不断地学习积累并输出，并在2015年5月份创立了微信公众号“大话存储”，继续总结和输出各类存储系统知识，皆为原创。本书即对这一年多来冬瓜哥的输出文章进行了整理再加工，并特意增加了30%的从未发布的额外内容。

如果说《大话存储》系列图书是一部系统性讲述存储系统底层的小说的话，那么本书相当于一部散文集，全篇形散神聚，自由穿梭于存储和计算机系统的底层和顶层世界中。其中的每一篇都表述了某个领域、课题或者技术，并围绕该技术展开叙述。冬瓜哥把全书划分为12个技术领域部分，每一个部分又包含多篇相关的文章。

其中有些文章中带有鄙人手绘的图片，为了保持原汁原味，决定保留原样，如果侮辱了你的审美观，请见谅。

阅读本书要求对存储系统有一定了解，最好是相当了解，否则会感到比较吃力。不过，吃力是好事，证明有提升空间，那就赶紧去买本《大话存储 终极版》看看正传吧，然后再来看后传。 当年冬瓜哥看一些文档的时候，也是很吃力，但是总感觉很有意思，也就坚持了下来。

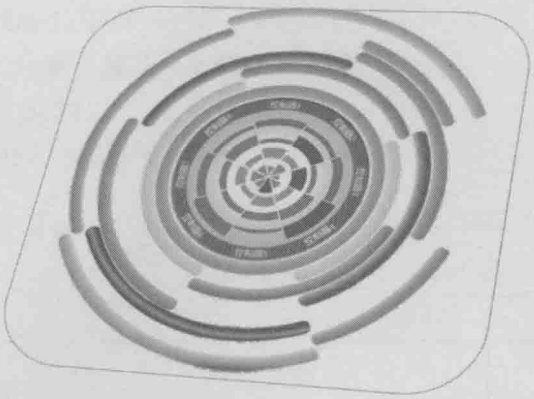
可能有人会想，后续会不会有《大话存储 外传》呢？嗯，或许吧，顺其自然！

最后，欢迎关注鄙人的微信公众号：

冬瓜哥

目录

第一章 灵活的数据布局



本章总结了冬瓜哥之前在存储系统产品设计方面的一点点成果，放在本书开头，作为一个开场白，也算是向各位存储界人士描述一下这几年“搞存储”的收效。

1.1 Raid1.0和Raid1.5

在机械盘时代，影响最终I/O性能的根本因素无非就是两个，一个是顶端源头，也就是应用的I/O调用方式和I/O属性；另一个是底端源头，那就是数据最终是以什么形式、状态存放在多少机械盘上的。应用如何I/O调用完全不是存储系统可以控制的事情，所以从这个源头来解决性能问题对于存储系统来讲是无法做什么工作的。但是数据如何组织、排布，绝对是存储系统重中之重的工作。

这一点从Raid诞生开始就一直在不断的演化当中。举个最简单的例子，从Raid3到Raid4再到Raid5，Raid3当时设计的时候致力于单线程大块连续地址I/O吞吐量最大化，为了实现这个目的，Raid3的条带非常窄，窄到每次上层下发的I/O目标地址基本上都落在了所有盘上，这样几乎每个I/O都会让多个盘并行读写来服务于这个I/O，而其他I/O就必须等待，所以我们说Raid3阵列场景下，上层的I/O之间是不能并发的，但是单个I/O是可以采用多盘为其并发的。所以，如果系统内只有一个线程（或者说用户、程序、业务），而且这个线程是大块连续地址I/O追求吞吐量的业务，那么Raid3非常合适。但是大部分业务其实不是这样，而是追求上层的I/O能够充分地并行执行，比如多线程、多用户发出的I/O能够并发地被响应，此时就需要增大条带到一个合适的值，让一个I/O目标地址范围不至于牵动Raid组中所有盘为其服务，这样就有一定几率让一组盘同时响应多个I/O，而且盘数越多，并发几率就越大。Raid4相当于条带可调的Raid3，但是Raid4独立校验盘的存在不但让其成为高故障率的热点盘，而且也制约了本可以并发的I/O，因为伴随着每个I/O的执行，校验盘上对应条带的校验块都需要被更新，而由于所有校验块只存放在这块盘上，所以上层的I/O只能一个一个

地顺着执行，不能并发。Raid5则通过把校验块打散在Raid组中所有磁盘上，从而实现了并发I/O。大部分存储厂商提供针对条带宽度的设置，比如从32KB到128KB。假设一个I/O请求读16KB，在一个8块盘做的Raid5组里，如果条带为32KB，则每块盘上的段（Segment）为4KB，这个I/O起码要占用4块盘，假设并发几率为100%，那么这个Raid组能并发两个16KB的I/O，并发8个4KB的I/O；如果将条带宽度调节为128KB，则在100%并发几率的条件下可并发8个小于等于16KB的I/O。

讲到这里，我们可以看到单单是调节条带宽度，以及优化校验块的布局，就可以得到迥异的性能表现。但是再怎么折腾，I/O性能始终受限在Raid组那少得可怜的几块或者十几块盘上。为什么是几块或者十几块？难道不能把100块盘做成一个大Raid5组，然后，通过把所有逻辑卷创建在它上面来增加每个逻辑卷的性能么？你不会选择这么做的，当一旦有一块盘坏掉，系统需要重构的时候，你会后悔当时的决定，因为你会发现此时整个系统性能大幅降低，哪个逻辑卷也别想好过，因为此时99块盘都在全速读出数据，系统计算xor校验块，然后把校验块写入热备盘中。当然，你可以控制降速重构，来缓解在线业务的I/O性能，但是付出的代价就是增加了重构时间，重构周期内如果有盘再坏，那么全部数据荡然无存。所以，必须缩小故障影响域，所以一个Raid组最好是几块或者十几块盘。这比较尴尬，所以人们想出了解决办法，那就是把多个小Raid5/6组拼接成大Raid0，也就是Raid50/60，然后将逻辑卷分布在其上。当然，目前的存储厂商黔驴技穷，再也弄出什么新花样，所以它们习惯把这个大Raid50/60组成“Pool”，也就是池，从而迷惑一部分人，认为存储又在革新了，存储依然生命力旺盛。

那冬瓜哥在这里也不妨顺水推舟忽悠一下，如果把传统的Raid组叫作Raid1.0，把Raid50/60叫作Raid1.5。我们其实在这里可以体会出一种周期式上升的规律，早期盘数较少，主要靠条带宽度来调节不同场景的性能；后来人们想通了，为何不用Raid50呢？把数据直接分布到几百块盘中，岂不快哉？上层的并发线程I/O在底层可以实现大规模并发，达到超高吞吐量。此时，人们被成功冲昏了头脑，没人再去考虑另一个可怕的问题。

至这些文字倾诸笔端时仍没有人考虑这个问题，至少从厂商的产品动向里没有看出。究其原因，可能是另一轮底层的演变，那就是固态介质。底层的车轮是不断地提速的，上层的形态是循环往复的，但有时候上层可能直接跨越式前进，跨越了其中应该有的一个形态，这个形态或者转瞬即逝，亦或者根本没出现过，但是总会有人产生火花，即便这火花是那么微弱。

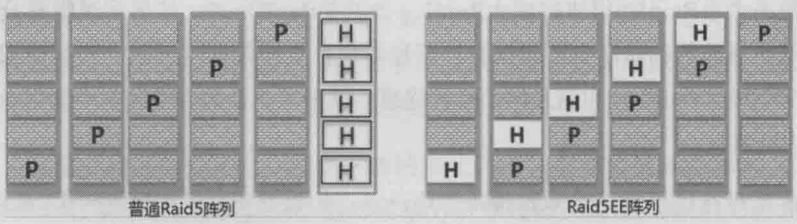
这个可怕的问题其实被一个更可怕的问题盖过了，这个更可怕的问题就是重构时间过长。一块4TB的SATA盘，在重构的时候就算全速写入，其转速决定了其吞吐量极

限也基本在80MB/s左右，可以算一下，需要58h，实际中为了保证在线业务的性能，一般会限制在中速重构，也就是40MB/s左右，此时需要116h，也就是5天5夜，我敢打赌没有哪个系统管理员能在这一周内睡好觉。

1.2 Raid5EE和Raid2.0

20年前有人发明过一种叫作Raid5EE的技术，其目的有两个，第一是把平时闲着没事干的热备盘用起来，第二就是加速重构。

很显然，如果把下图中用“H（hot spare）”表示的热备盘的空间也像校验盘一样，打散到所有盘上的话，就会变成图右侧所示的布局，每个P块都跟着一个H块。这样整个Raid组能比原来多一块磁盘可用于工作。另外，由于H空间也被打散了，当有一块盘损坏时，重构的速度理应被加快，因为此时可以多盘并发写入了。但是实际却不然，整个系统的重构速度其实并不是被这块单独的热备盘限制了，而是被所有盘一起限制了，因为热备盘以满速率写入重构后的数据的前提是，其他所有盘都以满速率读出数据，然后系统对其做xor。就算把热备盘打散，甚至把热备盘换成SSD、内存，对结果也毫无影响。



那到底怎样才能加速重构呢？唯一的办法只有像下图所示这样，把原本挤在5块盘里的条带，横向打散，请注意，是以条带为粒度打散，打散单盘是毫无用处的。这样，才能成倍地提升重构速度。