

ZStack 技术白皮书精选

快照和备份的区别

扫一扫二维码，获取更多技术干货吧



 ZStack中国社区@二群
扫一扫二维码，加入群聊。



长按扫码，关注ZStack官微

版权声明

本白皮书版权属于上海云轴信息科技有限公司，并受法律保护。转载、摘编或利用其它方式使用本调查报告文字或者观点的，应注明来源。违反上述声明者，将追究其相关法律责任。

摘要

大道至简·极速部署，ZStack 致力于产品化私有云和混合云。

ZStack 是新一代创新开源的云计算 IaaS 软件，由英特尔、微软、CloudStack 等世界上最早一批虚拟化工程师创建，拥有 KVM、Xen、Hyper-V 等成熟的技术背景。

ZStack 创新提出了云计算 4S 理念，即 Simple（简单）、Strong（健壮）、Smart（智能）、Scalable（弹性），通过全异步架构，无状态服务架构，无锁架构等核心技术，完美解决云计算执行效率低，系统不稳定，不能支撑高并发等问题，实现 HA 和轻量化管理。

ZStack 发起并维护着国内最大的自主开源 IaaS 社区——zstack.io，吸引了 6000 多名社区用户，对外公开的 API 超过 1000 个。基于这 1000 多个 API，用户可以自由组装出自己的私有云、混合云，甚至利用 ZStack 搭建公有云对外提供服务。

ZStack 拥有充足的知识产权储备，积极申报多项软著和专利，参与业内标准、白皮书的撰写，入选云计算行业方案目录，还通过了工信部云服务能力认证和信通院可信云认证。

ZStack 面向企业用户提供基于 IaaS 的私有云和混合云，是业内唯一一家实现产品化，并领先业内首家推出同时打通数据面和控制面无缝混合云的云服务商。选择 ZStack，用户可以官网直接下载、1 台 PC 也可上云、30 分钟完成从裸机的安装部署。

目前已有 1000 多家企业用户选择了 ZStack 云平台。

ZStack 实践汇 | 快照和备份的区别

作者：邵悠锋

0 前言

对于企业生产来说，最重要的就是数据。可以说数据是企业的生命，数据的破坏、丢失都会给企业造成巨大的损失，所以数据的保护至关重要。

数据的保护方案可以分为快照和备份，那么这两种方案有什么区别，又分别适用于什么场景呢？首先，我们需要明确一点，快照不是备份，当数据处于危险时，快照和备份可做的事情是不一样的，两种不同的机制是用来应对不同需求的。

1 快照

概括地说，快照是保存数据状态的方式，以便于能够及时恢复到该点。以下是存储网络工业协会（以下简称 SNIA）对快照的定义：

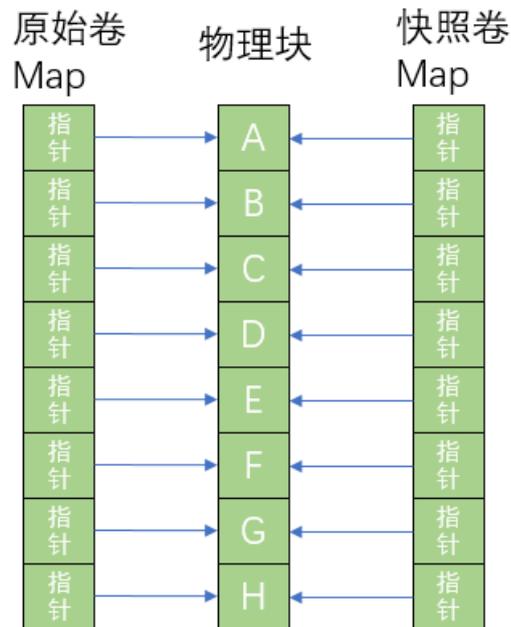
A type of point in time copy that preserves the state of data at an instant in time, by storing only those blocks that are different from an already existing full copy of the data

快照是一个特定时间点对数据状态的保护，只保存那些完整拷贝以外有变化的数据。换言之，如果数据没有变化，快照是不会保存额外数据的，快照可以看成是对某个特定时间点的数据的冻结。

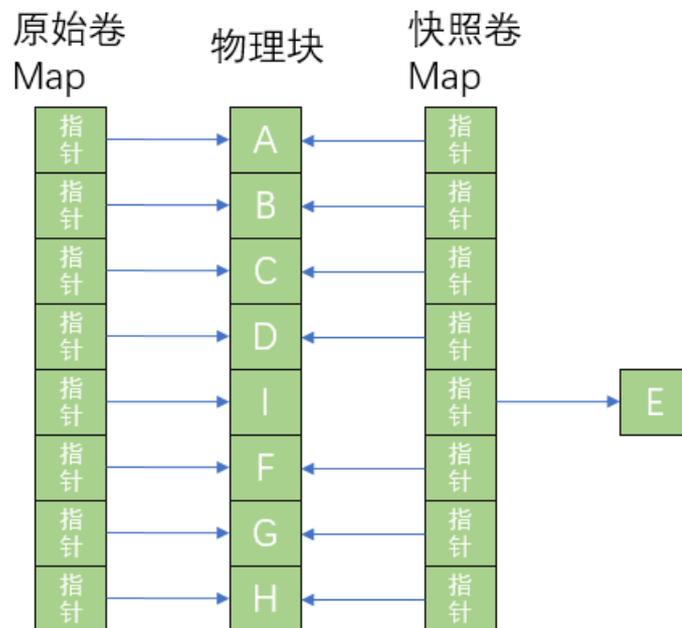
目前行业内快照实现有两种方式，COW（Copy On Write）和 ROW（Redirect On Write），这两种方式的区别是：当数据有变化时，对应数据块的处理方式也会不同。

1.1 COW

首先看 COW，快照生成后，如果没有数据的写入，那么快照卷的指针还是指向原始卷的数据块。

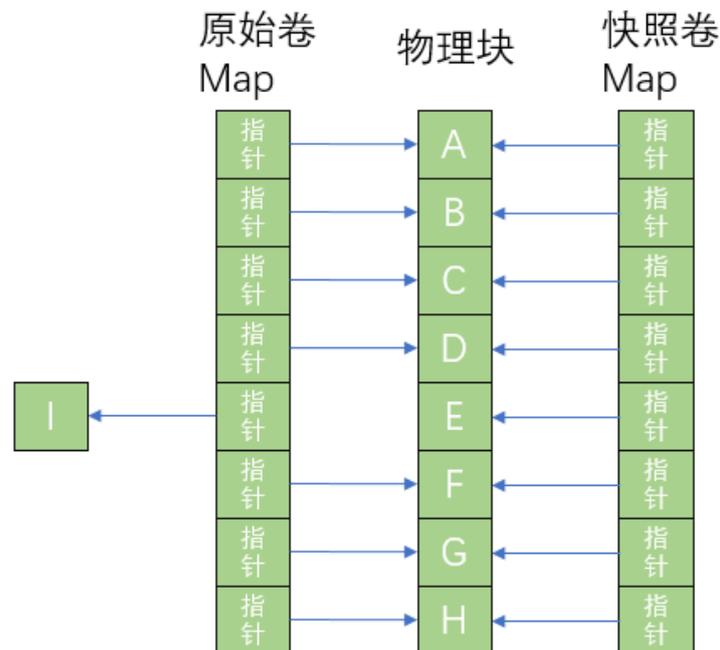


一旦有数据写入，比如写入此处的 E 数据块，系统会重新分配一个数据块，将 E 的数据整体拷贝到新的数据块中，然后原来 E 的位置写入新数据，计为 I。相应的，快照卷对 E 的指针需要指向新的数据块，而原始卷的指针指向是不变的。



1.2 ROW

再看 ROW，和 COW 的不同之处在于，如果有数据进行了修改，则分配数据块将修改的数据写入新的空间记为 I，然后将原始卷的指针指向 I，快照卷不做任何改动。



从以上的分析可以看到，COW 的优势在于：原始卷的物理块是连续的，没有碎片，缺点则是写放大了，本来 1 个写现在变成 3 个写，原始数据、快照指针指向、新数据，所以 COW 适合读多写少的业务；ROW 的优势是：写性能比 COW 更好，1 个写变成 2 个写，原始卷指针指向、新数据，但缺点是原始卷物理块不连续，会导致读性能较低，删除快照需要将原始卷指针修改，将数据提交至原始物理块，快照越大越消耗时间，故而 ROW 适合写密集型业务。

1.3 使用场景

从快照原理上可以看到，快照其实还是对原始卷有依赖，快照卷中大部分的指针依然会指向原始卷的数据块，如果有某个数据块损坏，很有可能快照是无法恢复的，因为有可能损坏的数据块就是没有变化过的。所以快照的作用是对一些操作的临时回滚，是一个临时手段。比如开发测试环境，以及生产环境做一些可能会影响操作系统、业务系统的操作，可以用快照做一个临时的保障，在异常时快速回滚到正常状态。

另外，快照如果创建的太多了，会极大的影响业务的性能，快照的各种链接会使得数据的读写变的非常复杂，同时也会占用大量的存储空间。所以，一般生产不建议使用快照，一定要使用快照的话，也是在进行危险的操作时，如可能损坏系统的操作系统更新或配置变动

一个典型的快照应用就是 windows 还原点，将系统的部分信息存放在 C 盘某个位置，并且不影响个人的文件，可以在系统发生故障时迅速还原。

2 备份

同样看下 SNIA 对备份的定义：

A collection of data stored on (usually removable) non-volatile storage media for purposes of recovery in case the original copy of data is lost or becomes inaccessible; also called a backup copy. To be useful for recovery, a backup must be made by copying the source data image when it is in a consistent state.

简单来说备份就是对数据的一个完整的拷贝，是新增加了一个一致性的副本，在原始卷因灾难或人为错误而损坏的情况下，能够为您提供恢复的可能性。与快照不同的是，备份独立于原始卷，可以轻松导出并存储到其他地方。

备份的类型可以分为全量备份、增量备份、差异备份三种。

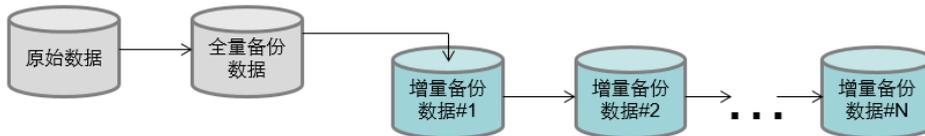
2.1 全量备份

全量备份顾名思义就是把整个数据都备份下来，全量备份的数据完整性最好，可以提供最好的数据保护，恢复速度也非常快，但是全量备份占用的空间非常大，可能会造成磁盘的浪费，而且因为是做所有数据的备份，数据量越大备份的时间就越长。



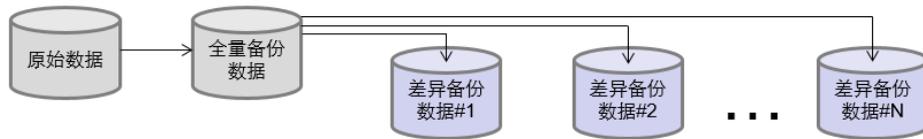
2.2 增量备份

增量备份是备份上一次备份之后发生变化的数据量，第一次做增量备份必然会做一个全量备份，后续所有的备份操作都是增量。增量备份会形成一条备份链，除了第一次的全量备份以外，后续所有增量备份都需要依赖于前一次备份的数据，如果一旦有某个中间的备份数据损坏或者丢失，那么在这个点以后的所有备份就全部失效了；如果要进行恢复，必须沿着从全量备份到指定时间点增量备份的顺序来进行恢复，这就使得恢复时间会变得很长，但是相对的，相比较于全量备份，每个增量备份的备份数据量非常小，只有上次备份以后发生变化的量，所以备份时间也非常短。



2.3 差异备份

差异备份是备份上一次全量备份之后发生变化的数据量，第一次做差异备份必然会做一个全量备份，后续所有的备份操作都是和全量备份相比发生变化的数据，在进行恢复时，只需对第一次全量备份和最后一次差异备份进行恢复。差异备份避免了全量备份和增量备份的缺陷，同时又具备了它们各自的优点。首先，它具有增量备份的备份时间短、节省磁盘空间的优势；其次，它又具有全量备份恢复时间短的特点，管理员只需要第一次全量备份和最后一次差异备份两份数据即可。差异备份的缺点是会存在一定量的重复数据，之前的差异备份数据都和最后一份差异备份数据有重复，当然可以根据设定的保留策略将多余的数据删除。



2.4 使用场景

从备份原理上可以看到，备份是拷贝出新的副本，是长久的数据保存方式，适用于对数据的长期保护，当数据发生错误或者丢失时，可以进行快速恢复，以保证业务的连续性。

其实备份过程中也用到了快照技术，一般要进行备份时都会做一个临时快照，把数据进行冻结，然后把快照的内容拷贝一份，最后删除快照。备份的时间点一般会选择凌晨业务量低谷的时候，以避免备份数据的不一致性。

3 快照和备份对比

一句话来对比：快照是数据存储的某一时刻的状态记录，备份则是数据存储的某一个时刻的副本，这是两种完全不同的概念。

来详细对比下快照和各种不同类型的备份方式的差异：

3.1 快照对比全量备份

通过之前的概念介绍，我们可以了解到：全量备份是对所有数据的一个拷贝，会将数据保存在不同的地方，和原始卷没有任何关系，是独立的存在；而快照还是依赖于原始卷，并且只有变化的数据块才会拷贝。

3.2 快照对比增量备份、差异备份

增量备份是将上次备份之后变化的数据拷贝出来，和原始卷是没有依赖关系的，但是和上次的备份之间有依赖关系，会有一条依赖链接，一直链接到上次的全量备份。

差异备份是将上次全量备份之后变化的数据拷贝出来，同样也是和原始卷没有依赖关系的，差异备份之间没有任何关系，但是和上次的全量备份之间有依赖。

对比 COW，COW 的快照之间没有依赖关系，但是都和原始卷有依赖，对于变化的数据块，会分配新的数据块将老的数据拷贝过去。

对比 ROW，ROW 的快照之间会有快照链，也和原始卷有依赖，当删除快照的时候，因为要把数据都提交给原始卷，所以会把此快照时间点之后的快照都一并删除。

4 总结

通过比较快照和备份的差异，可以看到其实快照是一种技术，而备份是一个过程，快照是某个时间点的状态，而备份是对数据的拷贝，快照并不能等同于备份。对于数据的临时保护，可以使用快照，如果要进行长期保护，则需要使用备份。

快照和备份都有各自的应用场景，在合适的场景使用恰当的方案，才能够最大程度的保护数据。