Ceph Rados

目录

** 版本修订记录 **

版本号	*修订时间*	*修订内容*
v1.0	*2018-08-13*	*初版修订*

一.简介

RADOS:Reliable,Autonomic Distributed Object Store.即可靠、自动化的分布式对象存储。是一个提供高可用、高可靠性、高性能分布式存储的架构，基于 RADOS 架构原理实现的 Ceph 系统，被称为“下一代存储“；

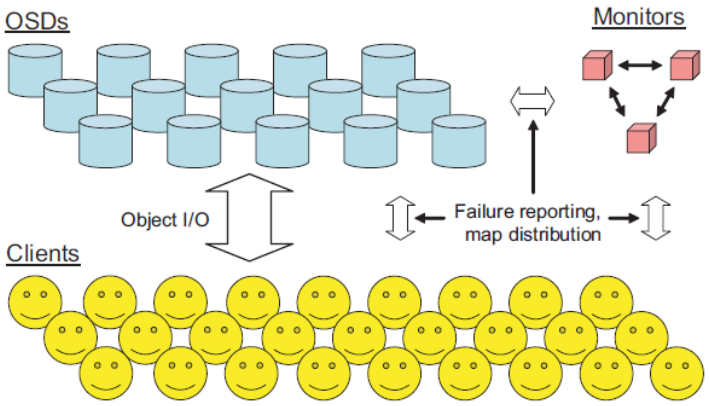

Figure 1 RADOWS 架构图

如上图 RADOS 架构图所示，一个 RADOS 系统中主要包括两个主要部分，对象数据存储集群可以包含大量的 OSD，主要用于监控 OSD 集群的小规模集群 Monitors，

二．集群管理

2.1 集群图

Monitors 管理集群的唯一方式是操作 cluster map(集群图).它指定了集群中的 OSD 间的关系、OSD 的状态以及确定了数据如何分布，集群图会被复制到 OSD 以及客户端中，每次 OSD 的状态发生改变或其他的事件导致数据分布发生变化时，集群图会发生改变，此时集群图的版本号(epoch)会增加，集群中的集群图的同步使用了增量惰性同步的方式，这主要时为了减轻 Monitors 的负载。同步的增量数据为两个相邻版本集群图的变化以及相差多个版本的数据变量的打包，同步的时机为周期性的 Monitor 和 Monitor 的信息交互、OSD 和 Monitor 通信或 OSD 和 OSD 之间通信时会较集群图版本的信息，集群图版本不一致时会触发同步；

2.2 数据放置

RADOS 中的数据对象分布是使用的伪随机的策略，当集群中增加新的存储节点时，为了保持集群中数据的均衡性，会有随机的部分数据迁移到新的存储节点上，这使得所有的设备的负载基本上保持均衡，另外也需要保证集群数据分布的稳定性，不能在添加或删除存储节点后导致所有的数据迁移。RADOWS 系统中数据的存放分为了两个计算过程：

(1) 数据对象首先被映射到 PG(Placement Group)中,PG 时对象集合的逻辑单位，每个 PG 的数据上的数据被分配到相对固定的 OSD 上，每个 OSD 上的数据一致的。每个对象放置到那个 PG 是有对象名称的哈希值决定的：

$$pgid = (r, hash(o) \& m)$$

其中：

- r: 表示复制集群中对象的复制因子；
- o: 表示对象的名称，在 Ceph 集群中，该值为文件名和条带序号的组合；
- m: 表示 PG 的数量，理论为 2**k -1,Ceph 中一般每个 OSD 上平均分配 100 个 PG；

(2) RADOS 系统使用 CRUSH 算法将 PG 映射到 r(复制因子)个 OSD，CRUSH 是一种稳定的分布式算法，能够根据 OSD 的容量或性能进行相对伪随机的 PG 映射，和 HASH 算法具有相似的功能；

以上可以看出基于计算的数据存储方式使得集群中取出了笨重、中心化的查询节点，增加了集群的扩展能力；

2.3 设备状态

OSD 会周期性向 Peering 的 OSD 发出心跳侦测周围的 OSD 是否发生了状态的变化，并在状态发生变化的情况下告知 Monitor,Monitor 收到报告后重新生成新的集群图。在 RADOS 集群中，OSD 的状态主要有一下几类：

- in: 代表该 OSD 处于集群中，可以被 PG 映射；
- out: 代表该 OSD 处于集群外，是不会被 PG 映射的；
- up: 代表 OSD 网络是可达的；
- down: 代表 OSD 进程可能关闭了；

集群中几种特殊的混合状态：

- down and in: 代表该 OSD 已经关闭了，但是该 OSD 上的数据并未被重新映射到其他的 OSD 上,Ceph 中该模式为降级模式(degreed mode)；
- up and out: 该状态代表 OSD 是可达的，但是该 OSD 当前并不会被 PG 映射，并且该 OSD 上的数据并不会发生迁移，一般在 OSD 重启或网络间接性故障的时候 OSD 会出现该状态；

2.4 Map 同步

由于 RADOWS 集群中可能包含了成千上万的 OSD，并且集群图的变化是家常便饭，导致将变化的集群图在集群中进行广播是不切实际的，这会导致集群中网络流量的大量消耗和网络带宽的无实际占用。集群图在 OSD 和 OSD 通信或客户端和 OSD 通信时才显得尤为重要，可以通过比较他们

的交互的消息来惰性的更新节点中的进群图，这有效的将 Monitor 分发任务负载转义到了各个 OSD 上。

每个 OSD 都会保存最近的集群图更新的增量数据，并用集群版本号 epoch 打上标签。当两个通信的 OSD 的集群图版本不一致时，会将更新的增量的数据进行同步，另外，集群中失效校验的心跳消息保证了集群图快速传输——在有 n 个 OSD 的时间复杂度为 O(log n)；

三.智能存储设备

数据分布的信息主要使用了集群图，这使得在集群中的 OSDs 可以分布式实施数据冗余策略、失败校验和失效恢复。

客户端在和 RADOWS 系统交互的时候，只需要将单个的写操作提交到第一个主 OSD(*primary OSD*)，它会负责所有副本数据的更新和一致性，这将对象复制相关的带宽转移到了存储集群的内部网络，简化了客户端的配置；结合对象版本和 PG 的短期日志，可以在节点间断性失效的情况下进行对象数据的快速恢复；

3.1 数据复制

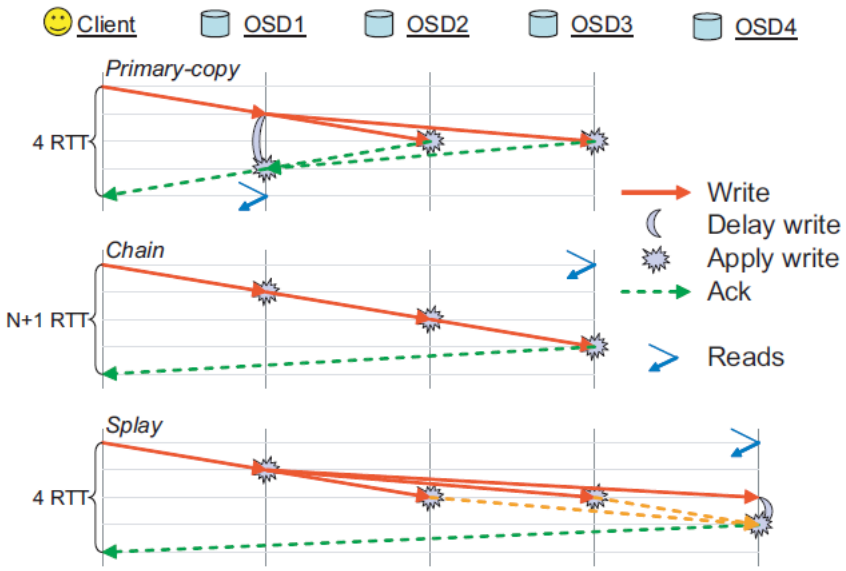

Figure 2 对象复制图

如上图所示，RADOWS 系统可以提供三种类型的数据复制：

- Primary-copy: 客户端和主 OSD 交互后，主 OSD 将数据并行的更行到其他副本 OSD，读写进程都在主 OSD 上；主 OSD 接收到所有副本 OSD 的写完成信号后，再向客户端返回写完成消息；

- Chain: 对象数据被先后顺序的写入各个 OSD；
- Splay:该种数据复制方式结合了 Primary-copy 的并行更新和 Chain 复制的读写分离；

3.2 数据一致性

所有的 RADOWS 消息，包括客户端产生的或来自其他 OSDs 的，都会被打上 map 版本的标记，这确保了以完全一致的方式进行所有的写入、读取和更新操作；如果一个客户端由于获取了旧版的集群图导致将数据发送给错误的 OSD，该 OSD 会比较客户端的集群图版本并将集群图版本的增量数据返回给客户端，这样它会更新后的集群图将数据重新定向到其他的 OSD 上；

3.3 失效校验

Peering 的存储节点间将会周期性的交换心跳信息来确保检测的设备的失效，在 TCP 套接字失败并尝试有限次的重连尝试后会见 OSD 的无法联通的状态发送给 Monitor,Monitor 收到后会将其标记为 down 状态。

3.4 数据迁移和恢复

当集群图更新或 PG 和 OSD 映射发生改变时，RADOWS 数据会发生迁移或失效恢复，集群图发生改变可有是由设备损坏、设备恢复、集群扩张或集群收缩、以及更换新的 CRUSH 分布式复制策略导致所有对象重分布引起。

在 RADOWS 系统中，使用了一种稳定的 peering 算法来为 PG 的内容建立一致性视图和、复制数据以及恢复数据的正常分布，这种策略依赖于 OSD 主动复制 PG 日志和 PG 当前的内容纪录这个接本设计前提，甚至对象的数据还有可能产生局部缺失，因此，尽管恢复过程是缓慢的，并且对象有时候处于降级模式，但是 PG 的元数据是被安全的保存的，这简化了恢复算法的设计并允许系统可靠的检测的数据的丢失

- Peering:

当 OSD 收到集群图更新时，它会遍历所有的增量更新，之后它会检查并有可能调整 PG 的状态，任何本地存储的 PG 的活动 OSD 列表的变化都必须 re-peer.和复制一样，peering 过程正对集群中的各个 PG 来说都是独立的；

Peering 过程由主 OSD 发起，对于一个 PG 中不是主 OSD 的 OSD,都会发送通知消息给主 OSD，这个消息包括了：本地存储的 PG 的基本状态、最近的更新、一定范围内的 PG 日志、以及其最近的集群图 epoch;

- Recovery

Declustered replication 一个优势是可以进行并行的故障恢复；在 Peering 过程中，通过非主 OSD 的通知的消息就知道了 replica 缺失的对象数据，它可以将任何对象推送给 Replica OSD;

四．Monitor

Monitors 是一个小的集群，主要维护集群图的主副本，并周期性的更新配置或 OSD 状态的改变，Monitor 集群是基于 Paxos 算法的，旨在支持可用性和更新延迟的一致性和持久性。需要注意的是,集群中需要有大多数 Monitor 是可用的;

4.1 Paxos Service

4.2 工作负载和扩展性

五．参考资料

【1】RADOS: A Scalable, Reliable Storage Service for Petabyte-scale Storage Clusters

六．附录

6.1 API