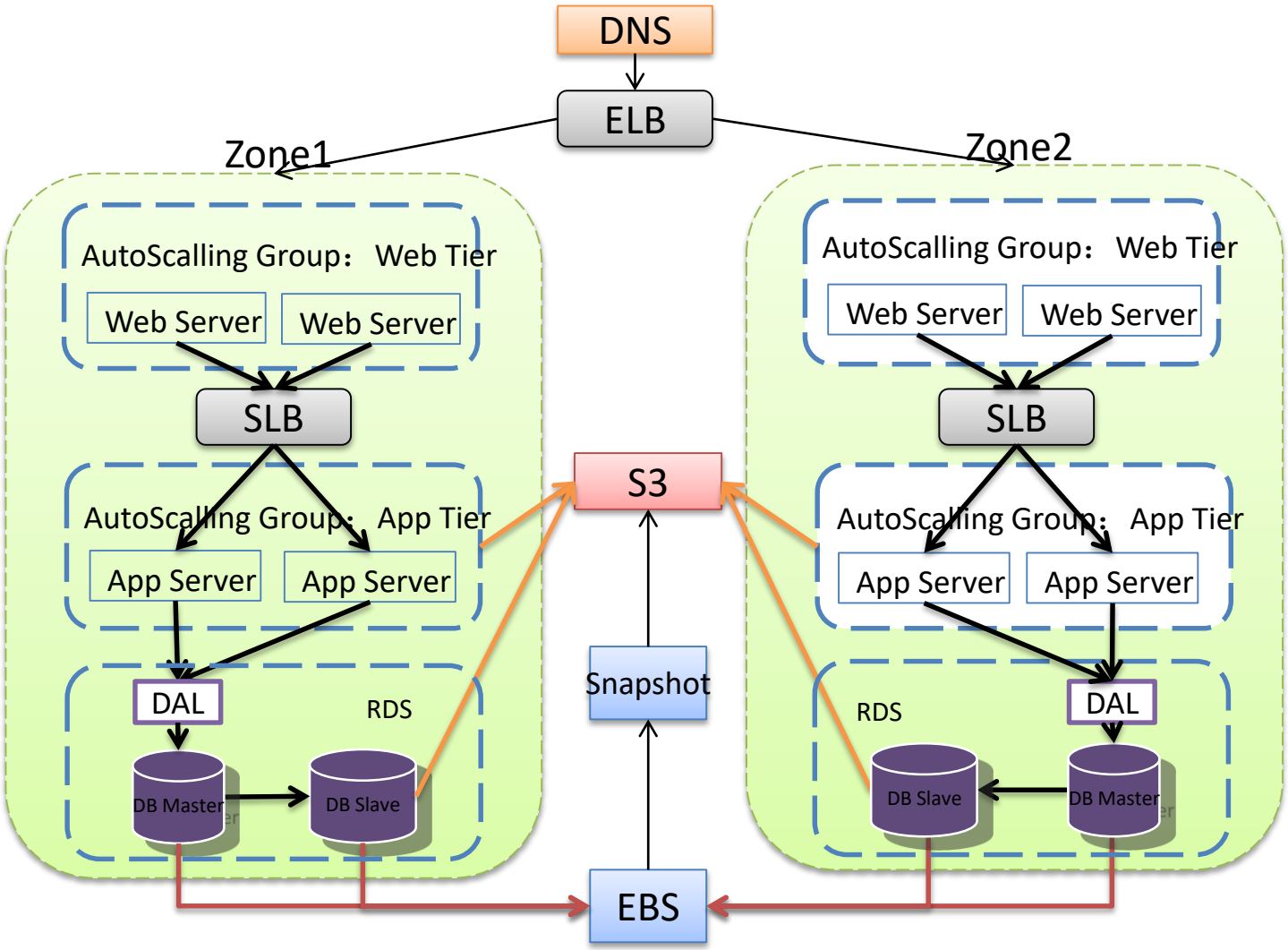
EBS设计方案

京东云计算中心 京东虚拟化总监 何雨

云计算部 2013.4

目录

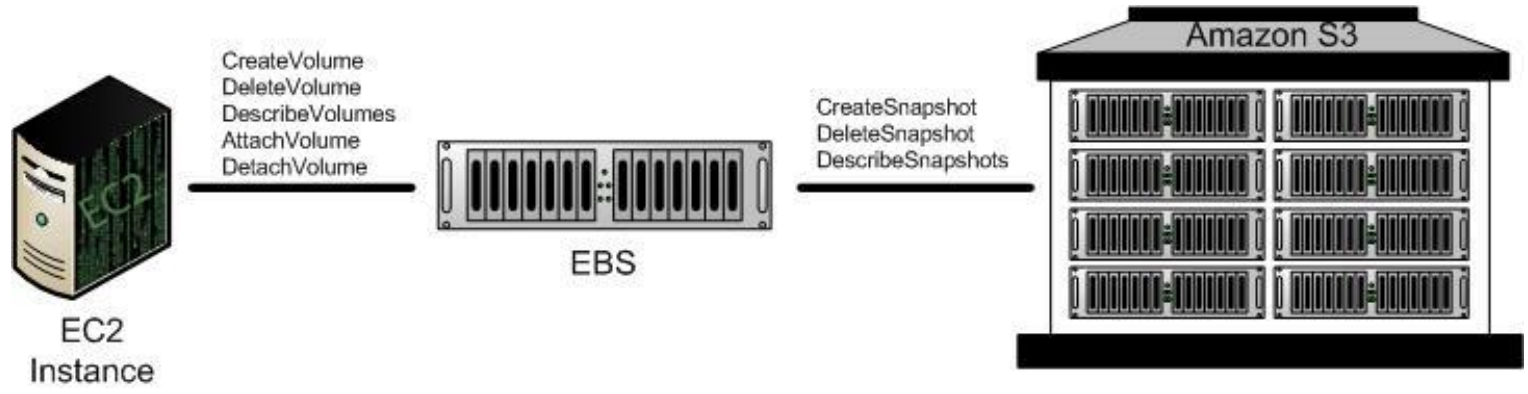
EBS在IaaS架构中的位置



什么是EBS

EBS的概念来源于Amazon，EBS全称是Elastic Block Store （弹性块存储）

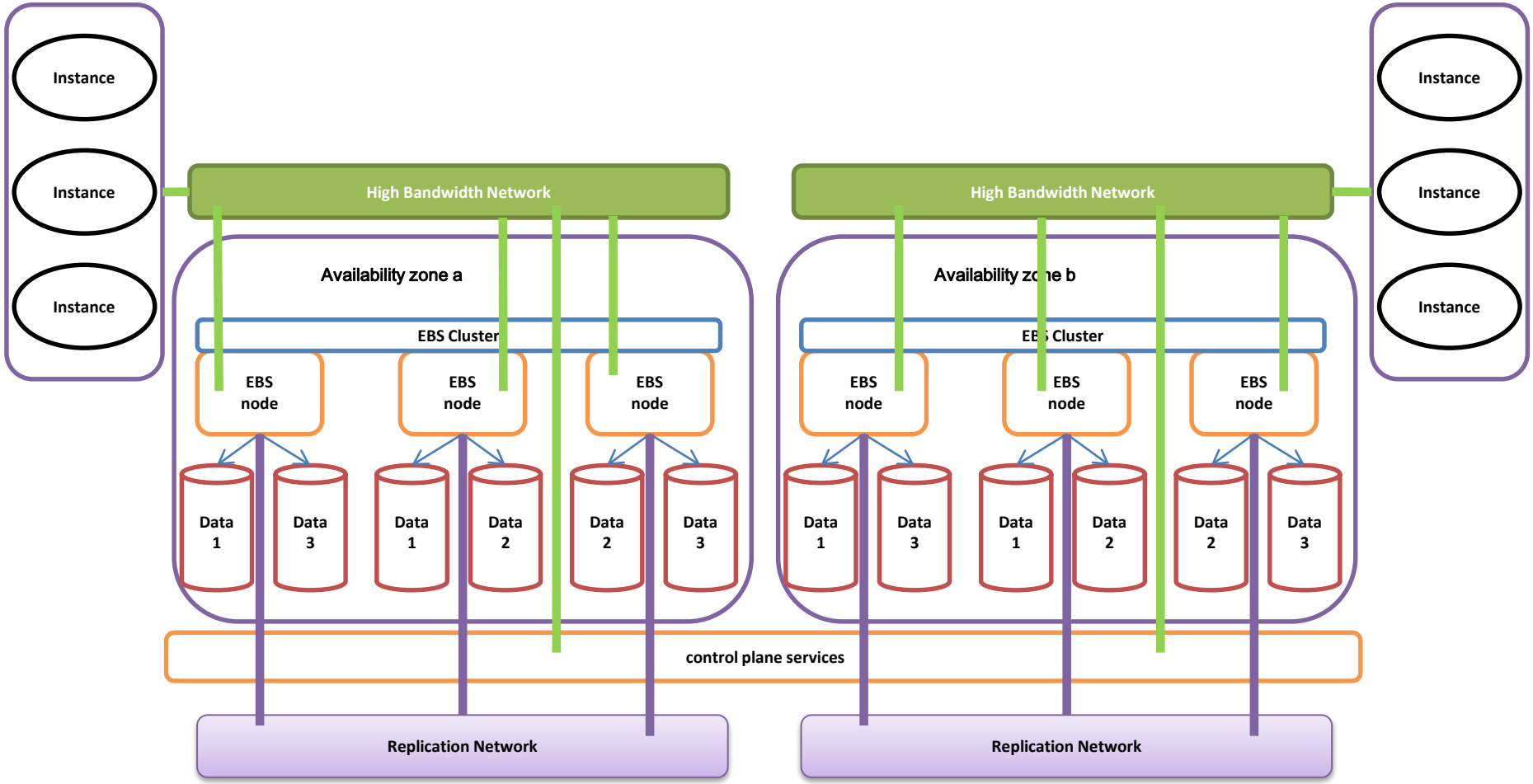
你可以在虚拟机实例上通过EBS分配一个虚拟硬盘空间给MySQL用，将信息持久到EBS上



EBS的设计需求

- EBS要给虚拟机实例提供高可用、高可靠的块级volume
- EBS应适合于一些需要访问块设备的应用，比如数据库、文件系统等
- EBS volume像裸块设备一样，有块设备接口，可以在volume上创建文件系统
- 每个volume要有冗余机制，保证当单个硬件失效时，数据不会丢失
- EBS可以创建当前时间点的快照，这些快照可用于创建新的volume

EBS的架构

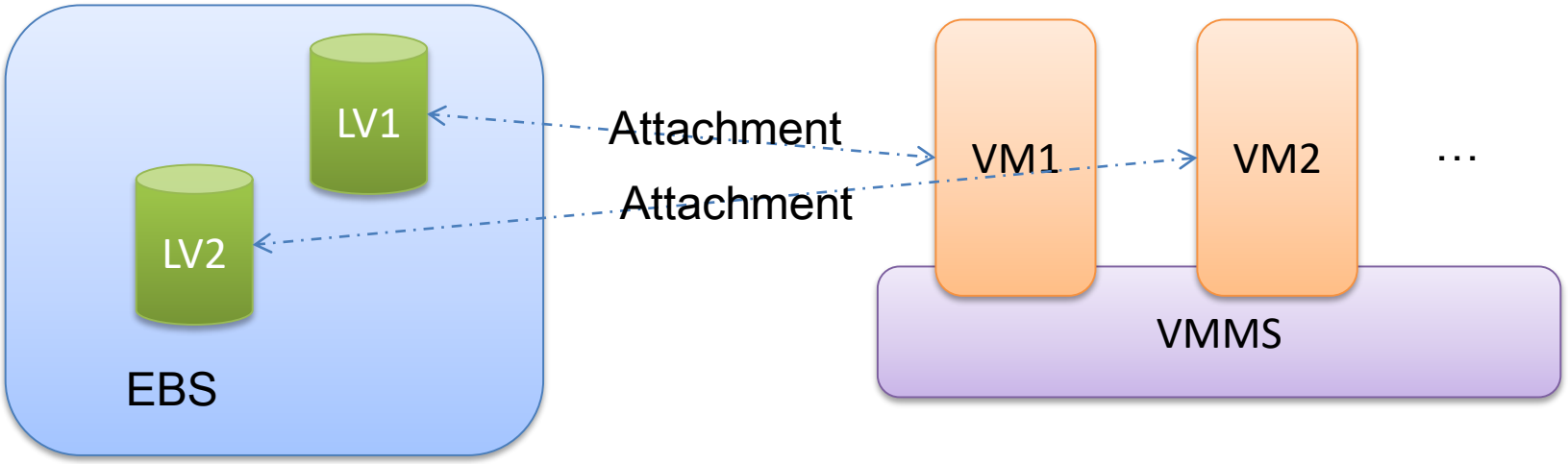

Instance
Instance
Instance
High Bandwidth Network
Availability zone a
EBS Cluster
EBS node
EBS node
EBS node
Data 1
Data 3
Data 1
Data 2
Data 2
Data 3
High Bandwidth Network
Availability zone b
EBS Cluster
EBS node
EBS node
EBS node
Data 1
Data 3
Data 1
Data 2
Data 2
Data 3
Instance
Instance
Instance
control plane services
Replication Network
Replication Network

给虚拟机实例提供块级volume

在EBS中创建多个逻辑卷（LV1、LV2……）

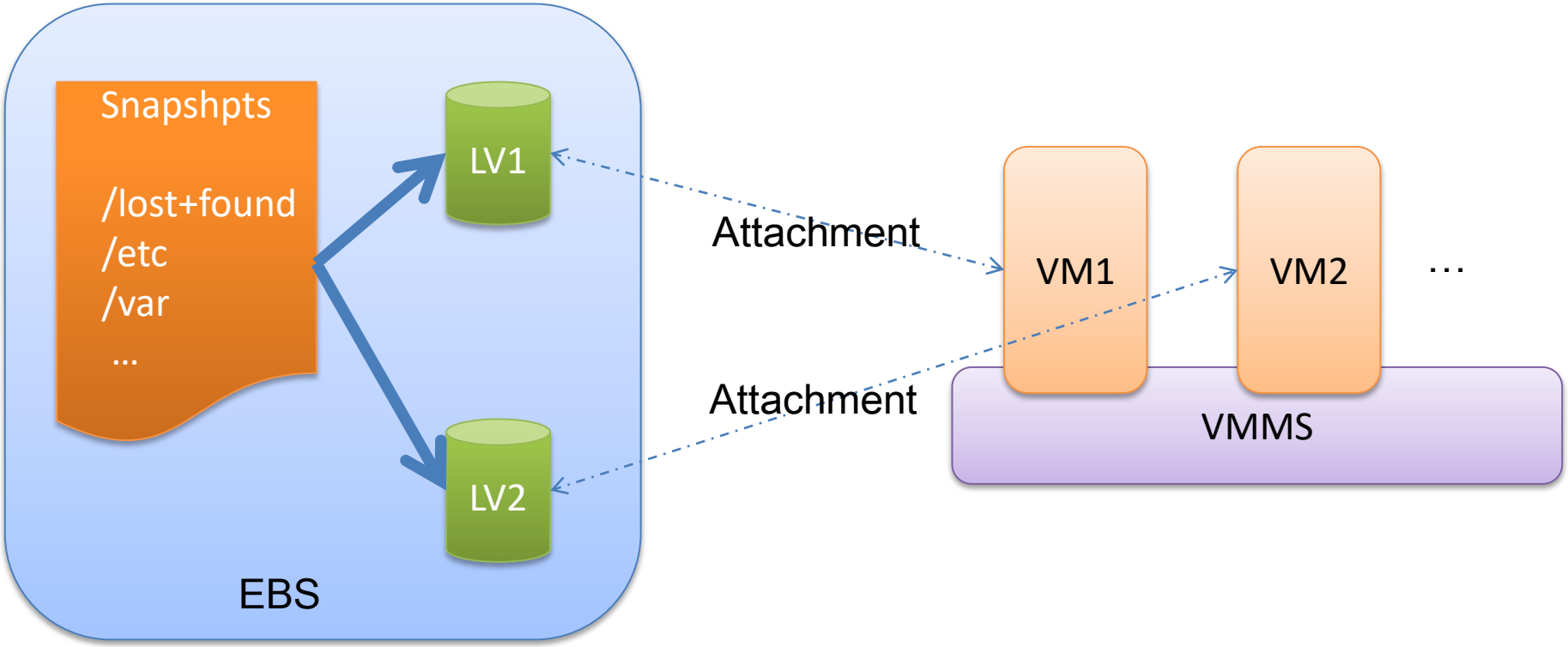
在VMMS（虚拟机管理系统）上启动多个VM instance（虚拟机实例）

每个逻辑卷分别Attachment到相应的虚拟机实例上，作为该虚拟机的存储设备，就好像是虚拟机的本地硬盘一样



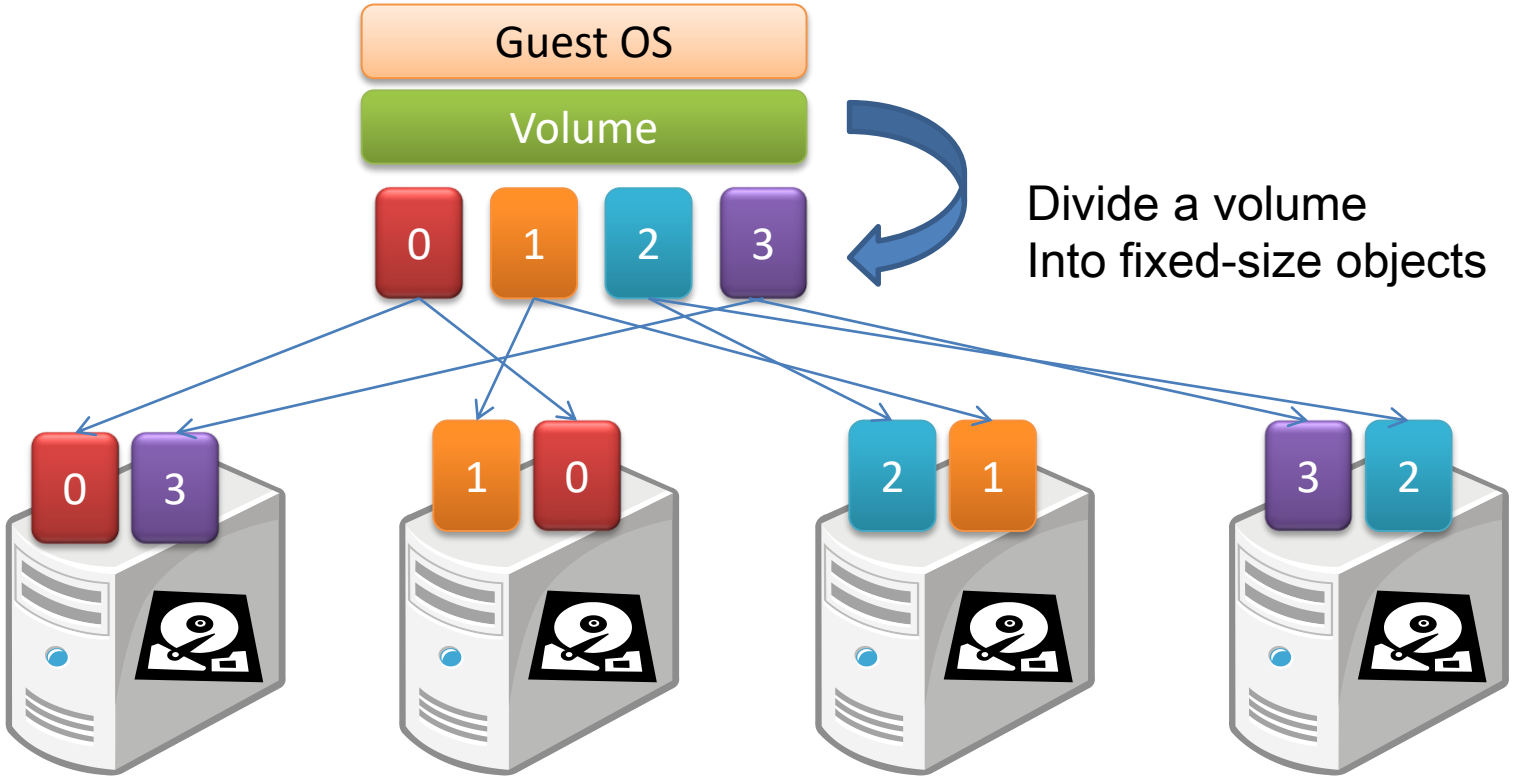
以快照为基础进行Volume创建

逻辑卷可以以某个时间点的快照为基础进行创建，这样可以很容易的创建多个相同的逻辑卷，使得多个虚拟机实例可以执行相同的计算任务



Volume的冗余机制

Volume切分成多个objects，分散存储在多个节点上，每个objects都有相应的副本



EBS的开源现状

目前开源里，就Ceph和sheepdog能实现EBS的功能。

目前sheepdog，只能支持kvm。

Ceph同时支持文件存储、对象存储及块存储。

Dreamhost一直都在测试Ceph，推出了基于ceph的对象存储服务

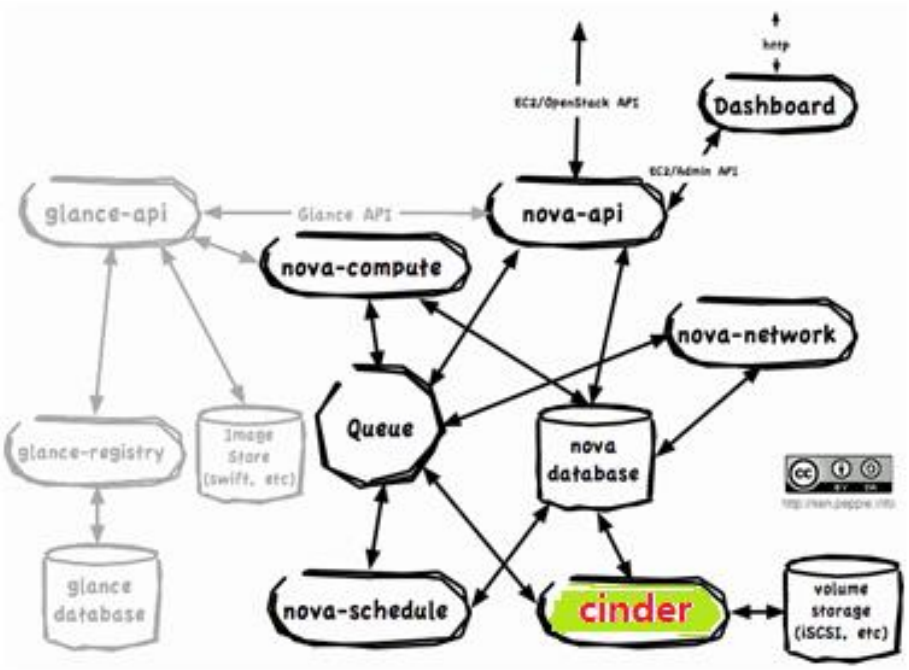
近日， Inktank和SUSE公司宣布战略合作，联合为作为SUSE Cloud一部分的Ceph Distributed Storage System提供企业级支持

ceph的RBD已经可以作为OpenStack的后端存储

目前的OpenStack Cinder的核心成员里有Ceph的开发者参与。

OpenStack Cinder对EBS的支持

- cinder是一个资源管理系统、负责存储资源的分配
- cinder把不同的后端存储进行封装，向外提供统一的API
- openstack没有开发块设备存储系统，cindre只是结合不同后端存储的driver提供块设备存储服务



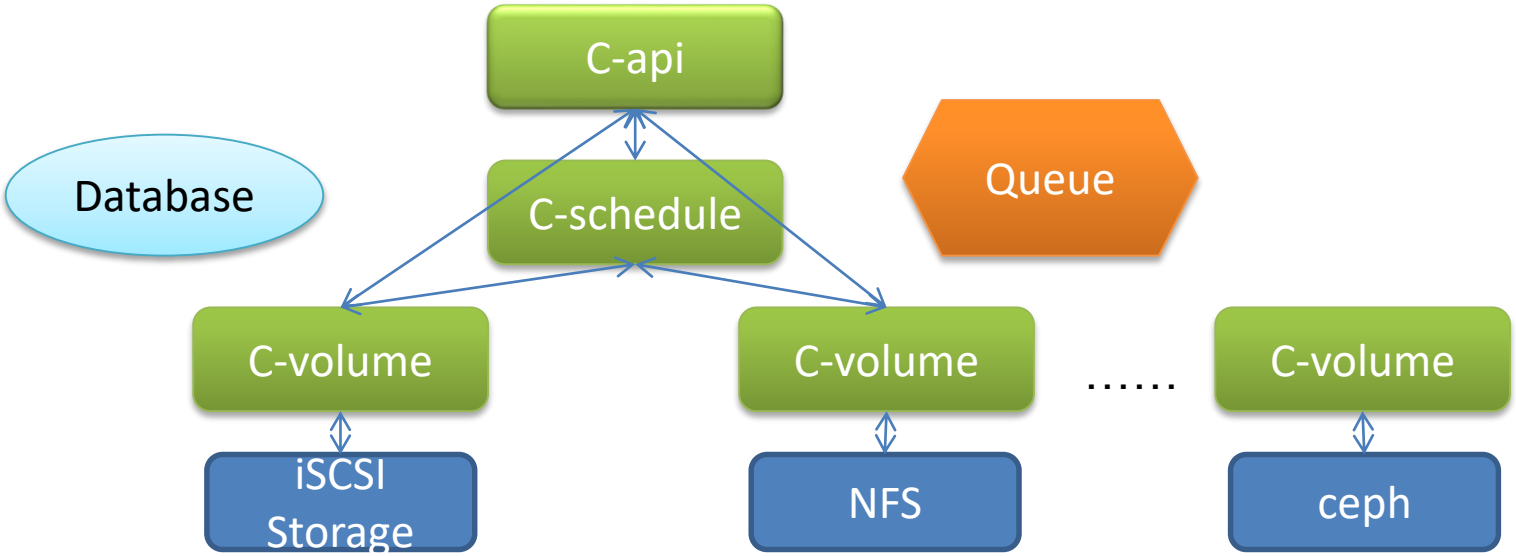
Cinder架构

目前cinder的架构和nova的一样，非常方便扩展

- Api组件负责向外提供REST接口
- schedule组件负责分配存储资源
- volume组件负责封装driver，不同的driver负责控制不同的后端存储。

组件之间的RPC靠消息队列实现。

cinder的开发工作主要集中在schedule和driver，以便提供更多的功能，支持更多的后端存储。



Cinder后端EBS存储的选择

如何选择后端存储，可以从性能、可靠性、价格三方面考虑

假如本来就有IP-SAN或NAS，而且cinder也支持，则就可以直接用, SAN和NAS的缺点(价格高、数据与计算分离、扩展性差、带宽有限制)

GlusterFS作为块设备存储的缺点(不在kernel、使用fuse，速度慢)

对成本敏感的也可以选择iSCSI+LVM+RAID的方案

要综合解决虚拟机存储的安全，成本，性能的问题，也可以用ceph和sheepdog

目前sheepdog，只能支持kvm

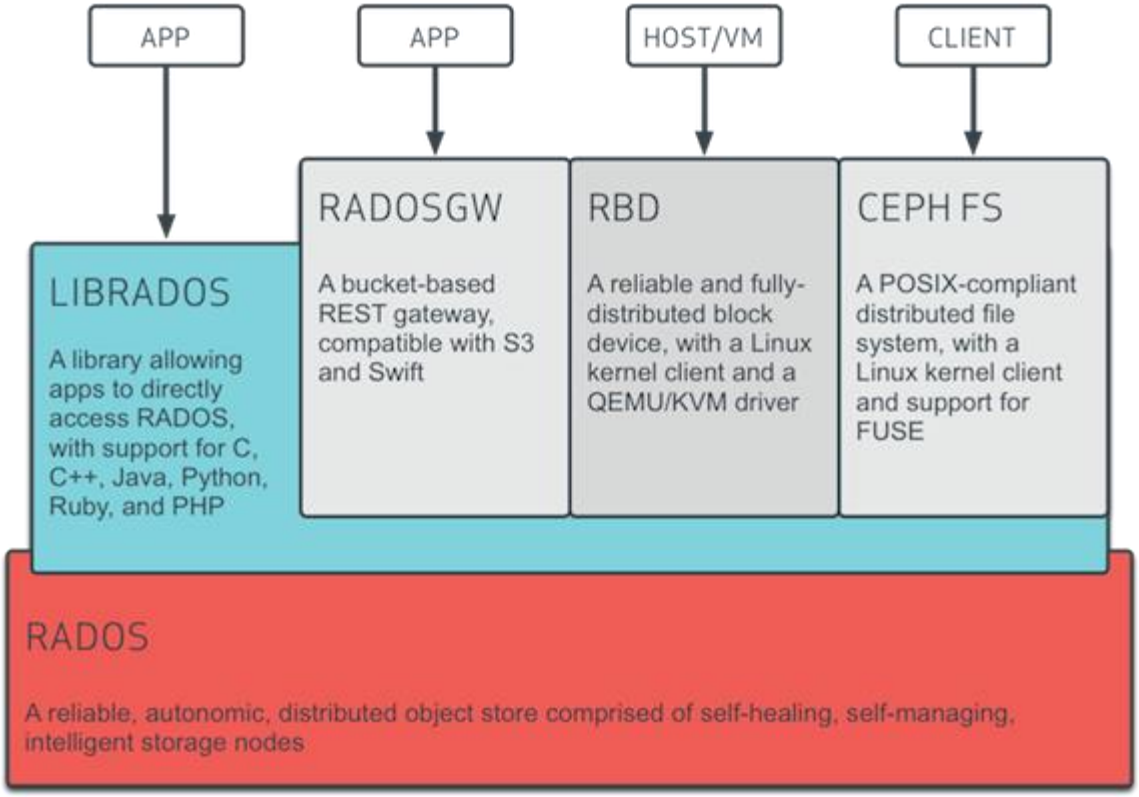
ceph的优点(在主线kernel中、client运行在内核态、支持普通硬件、线性扩展)

块设备存储解决方案必备条件(线性扩展、自动化)

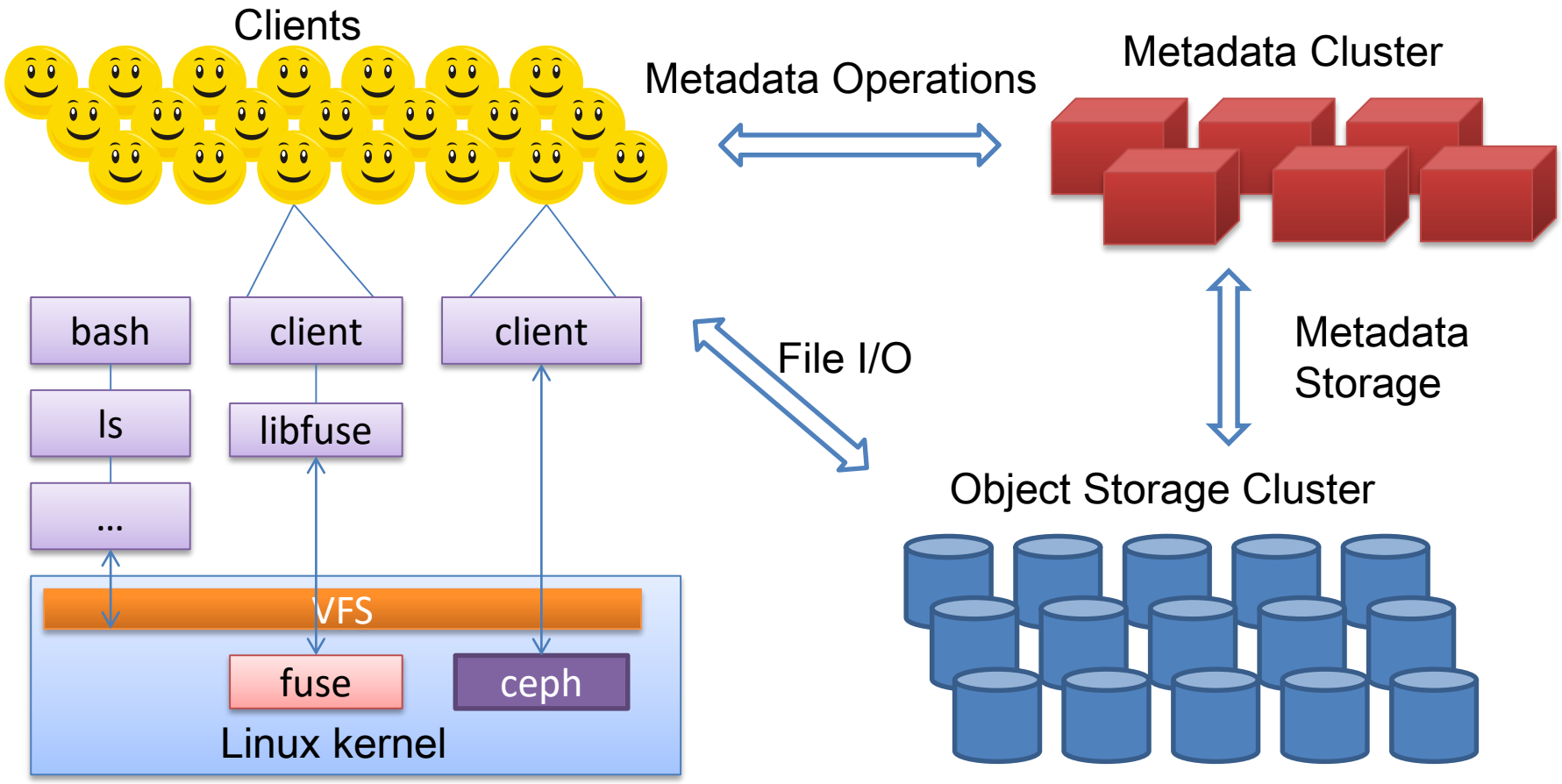
iSCSI	NFS(NAS)	其他
•IET+LVM/TGT+LVM •Solaris •HP Lefthand •IBM XIV/StorwizeSVC •Nexenta •Dell EqualLogic •SolidFire •NetApp	•NetApp •Glusterfs	•Ceph •Sheepdog

Ceph的统一存储

Ceph uniquely delivers object, block, and file storage in one unified system



Ceph架构

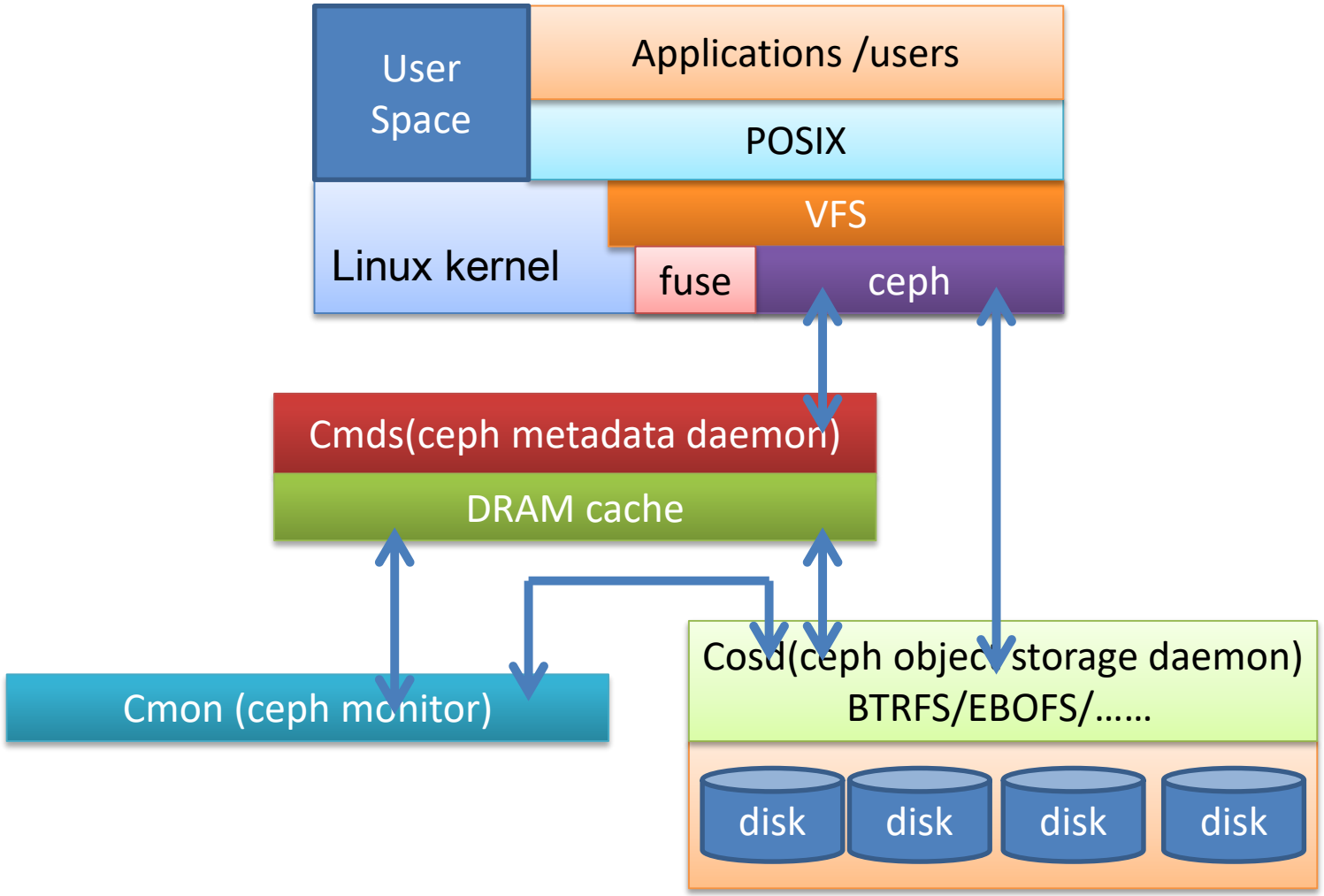

Clients
Metadata Operations
Metadata Cluster
bash
client
client
ls
libfuse
...
File I/O
Metadata
Storage
Object Storage Cluster
VFS
fuse
ceph
Linux kernel

Ceph Client

Ceph Client 对用户是透明的，为用户提供访问接口

早期版本的Ceph 利用FUSE，在用户空间实现访问接口，很大程度上简化其开发

今天，Ceph 已经被集成到主线内核，使其访问更加快速



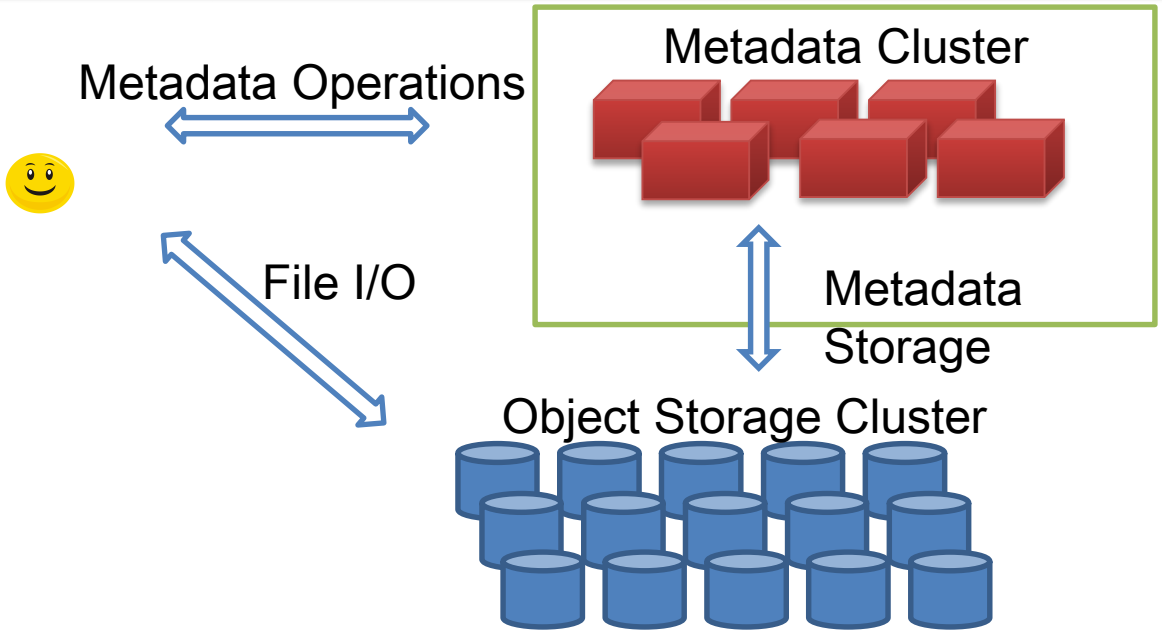
Ceph Metadata servers

元数据服务器管理文件系统的名称空间。元数据和数据两者都存储在对象存储集群，但两者分别管理，支持可扩展性

每个元数据服务器的主要应用就是一个智能元数据缓存（因为实际的元数据最终存储在对象存储集群中）

进行写操作的元数据被缓存在一个短期的日志中，它最终还是被推入物理存储器中。这个日志对故障恢复也很有用：如果元数据服务器发生故障，它的日志就会被重放，保证元数据安全存储在磁盘上

元数据服务器将文件名转变为索引节点，文件大小，以及Ceph 客户端用于文件 I/O 的分块数据



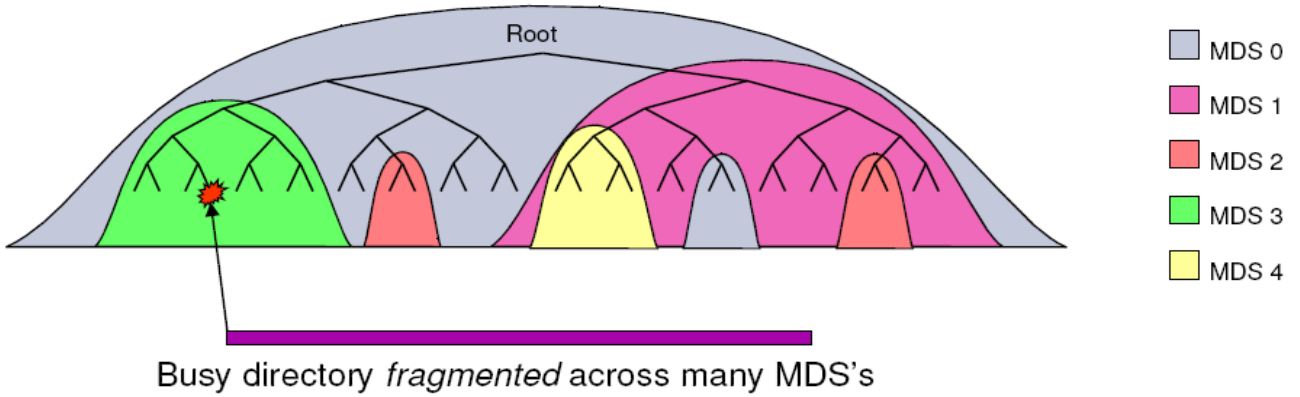
Ceph Metadata Partition

Scalability

- Arbitrarily partitioned metadata

Adaptability

- Cope with workload changes over time, and hot spots



MDS能够自适应地复制和分配名称空间，避免出现热点。

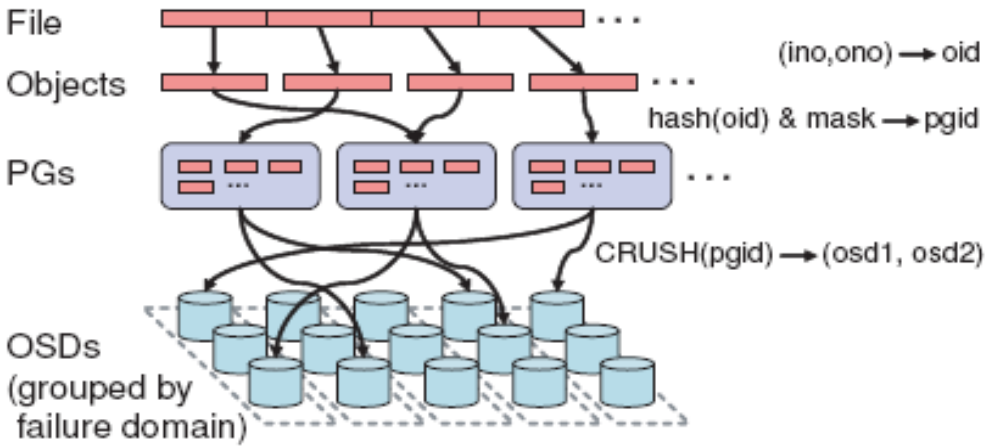
MDS管理的名称空间可以（为冗余和性能）进行重叠。

MDS到名称空间的映射使用动态子树逻辑分区执行，它允许 Ceph 对变化的工作负载进行调整（在元数据服务器之间迁移名称空间）。

Ceph 文件映射CRUSH

Ceph消除了文件的数据块映射表，通过计算获取块的存储位置，而不是查表

- 一个文件被分配到一个来自MDS的 inode number（INO）的唯一的标识符, 然后文件被分割成一些对象。
- 使用 INO 和 object number（ONO），每个对象都分配到一个对象 ID（OID）。在 OID 上使用一个简单的哈希，每个对象都被分配到一个放置组（PGID），它是一个对象的概念容器。
- 最后，放置组到对象存储设备的映射是一个伪随机映射，使用一个叫做 *Controlled Replication Under Scalable Hashing*（CRUSH）的算法。这样一来，放置组（以及副本）到存储设备的映射就不用依赖任何元数据，而是依赖一个伪随机的映射函数。这种操作是理想的，因为它把存储的开销最小化，简化了分配和数据查询。



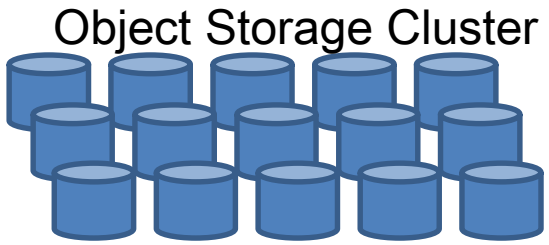
Ceph存储服务器集群

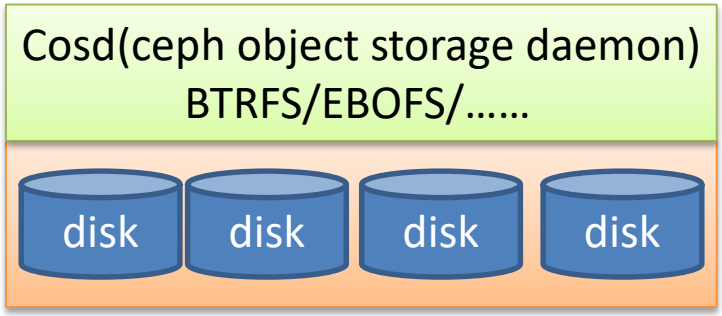
Ceph 对象存储设备执行从对象到块的映射。这个动作允许本地实体以最佳方式决定怎样存储一个对象

Ceph 的早期版本在本地存储器上实现一个自定义低级文件系统名为 EBOFS， EBOFS实现一个底层存储的非标准接口。今天，B-tree 文件系统（BTRFS）可以被用于存储节点

因为 Ceph 客户端实现 CRUSH，而且对磁盘上的文件映射块一无所知，存储设备就能安全地管理对象到块的映射，允许存储节点复制数据（当发现一个设备出现故障时）

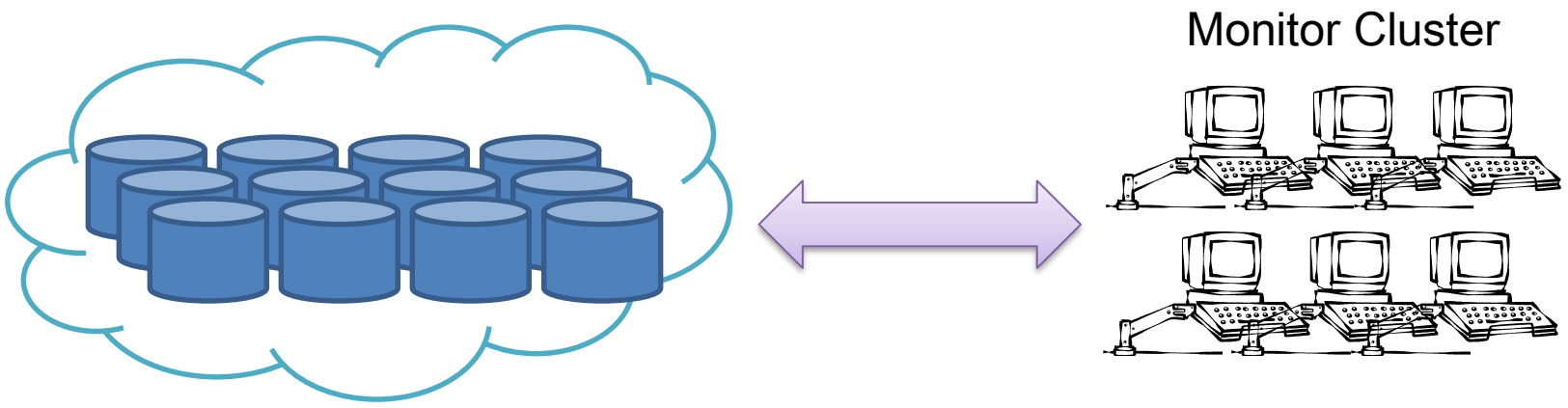
分布的故障恢复也允许存储系统扩展，因为故障检测和恢复可以跨域整个系统，Ceph 称其为 RADOS（Reliable, Autonomic Distributed Object Store）





Ceph 监视器

- Ceph 包含实施集群映射管理的监视器
- 当对象存储设备发生故障或者新设备添加时，监视器就检测和维护一个有效的集群映射
- Ceph 使用 Paxos，它是一种解决分布式一致性问题的算法



Ceph同时支持文件存储、对象存储及块存储

Block Devices

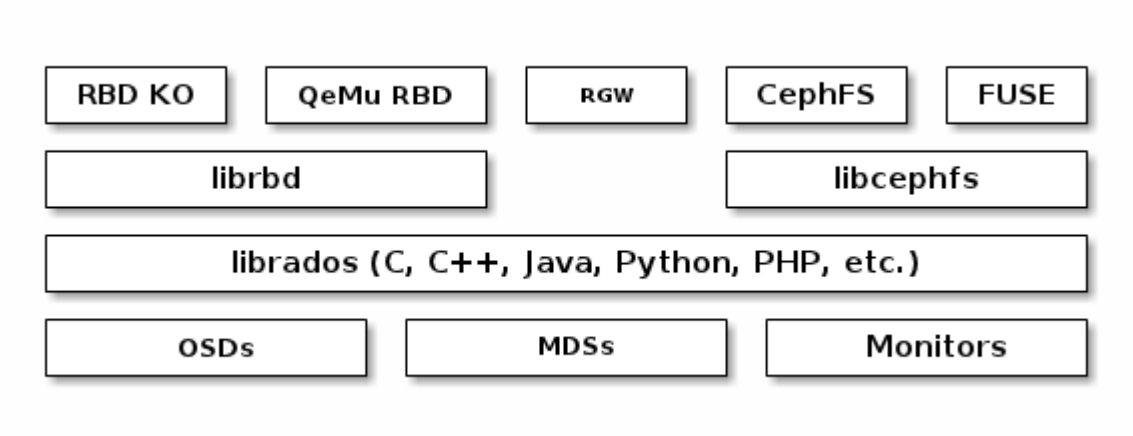
- The RADOS Block Device (RBD) service provides resizable, thin-provisioned block devices with snapshotting and cloning. Ceph stripes a block device across the cluster for high performance. Ceph supports both kernel objects (KO) and a QEMU hypervisor that uses librbd directly–avoiding the kernel object overhead for virtualized systems.

RESTful Gateway

- The RADOS Gateway (RGW) service provides RESTful APIs with interfaces that are compatible with Amazon S3 and OpenStack Swift.

Ceph FS

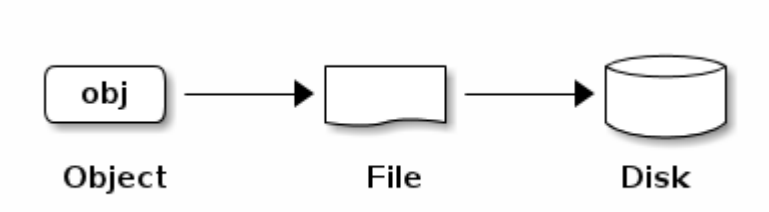
- he Ceph Filesystem (CephFS) service provides a POSIX compliant filesystem usable with mount or as a filesytem in user space (FUSE).



Ceph Object的数据格式

Each object corresponds to a file in a filesystem, which is typically stored on a single storage disk.

OSDs store all data as objects in a flat namespace (e.g., no hierarchy of directories). An object has an identifier, binary data, and metadata consisting of a set of name/value pairs

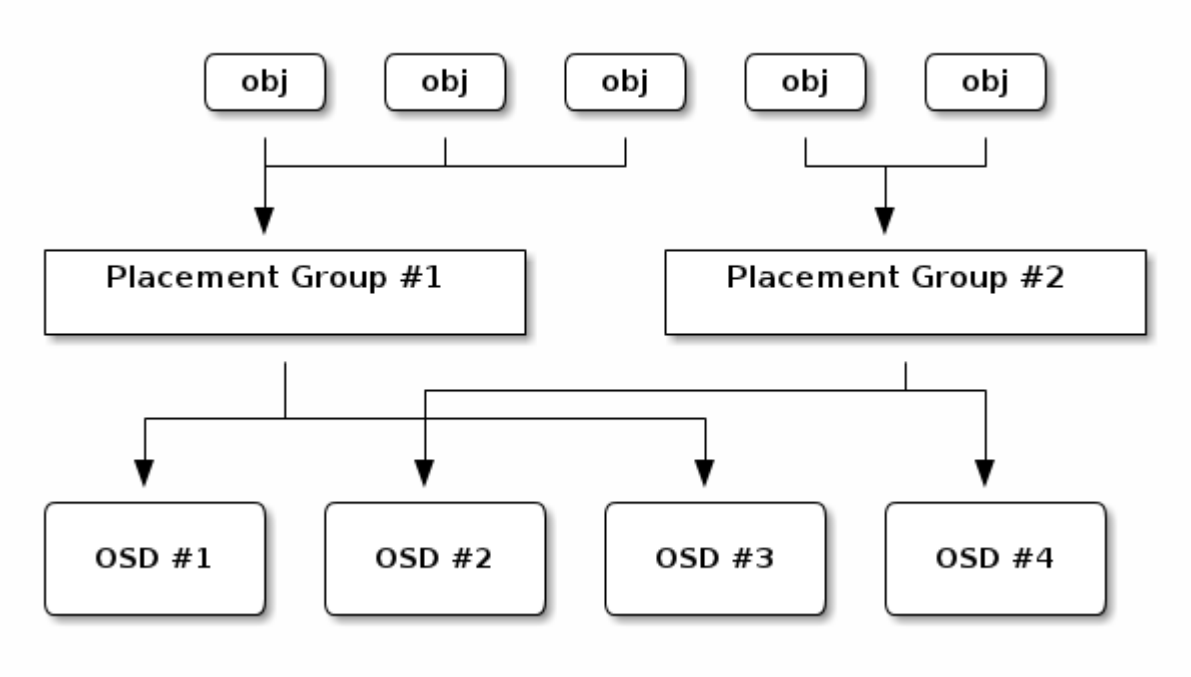


ID	Binary Data	Metadata
1234	01010101010101001101010010 01011000010101001101010010 01011000010101001101010010	name1 value1 name2 value2 nameN valueN

Ceph Object与OSD的松耦合

Mapping objects to placement groups instead of directly to OSDs creates a layer of indirection between the OSD and the client

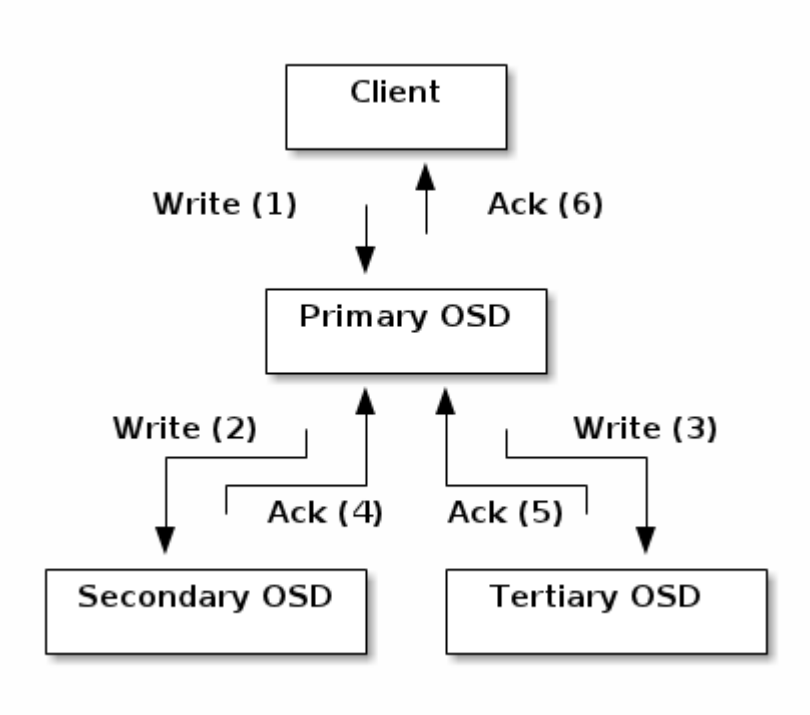
This layer of indirection allows Ceph to rebalance dynamically when new OSDs come online.



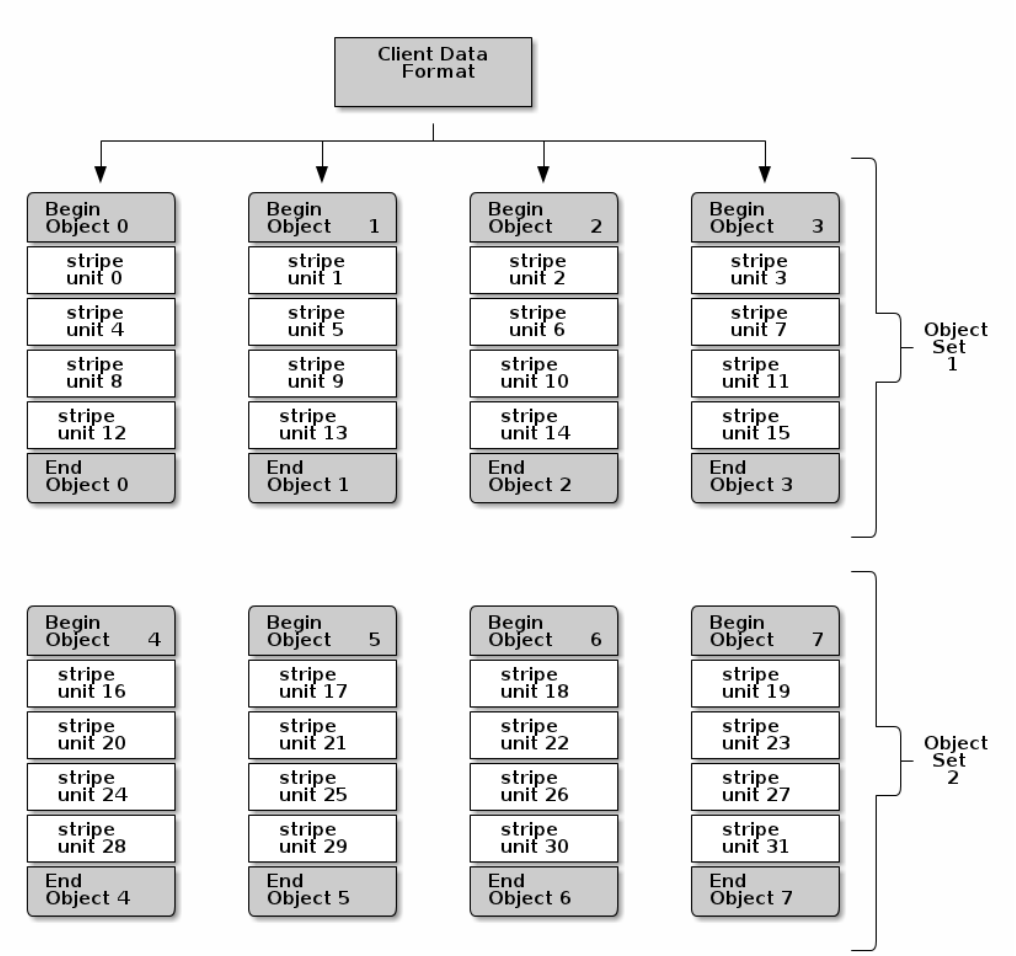
Ceph Client的写流程

The client writes the object to the identified placement group in the primary OSD

Then, the primary OSD with its own copy of the CRUSH map identifies the secondary and tertiary OSDs for replication purposes, and replicates the object to the appropriate placement groups in the secondary and tertiary OSDs (as many OSDs as additional replicas), and responds to the client once it has confirmed the object was stored successfully.



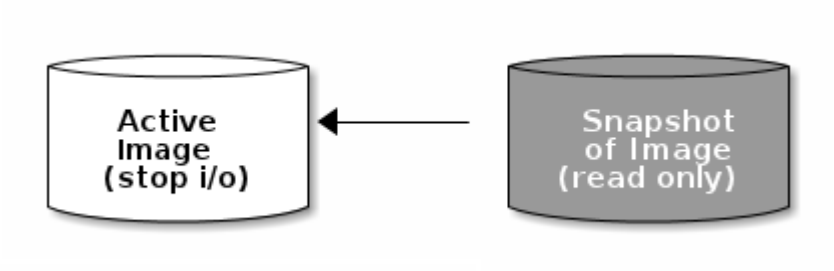
Ceph 存储数据的条带化

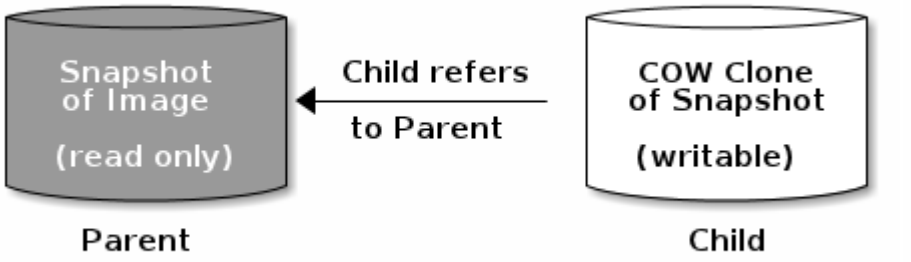


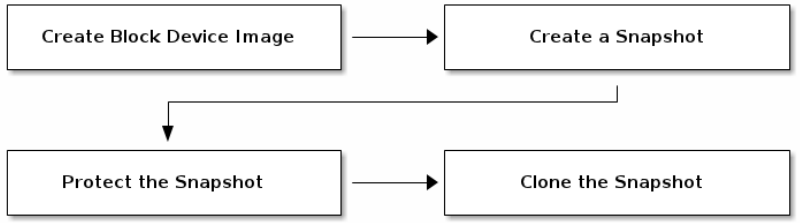
Ceph Snapshot

STOP I/O BEFORE snapshotting an image

Snapshot layering enables Ceph block device clients to create images very quickly







结束
谢谢