



新一代分布式存储系统 Curve

D I G I T A L S A I L

李小翠

网易数帆存储团队

概述



Curve 是高性能、高可用、高可靠的分布式存储系统

- 高性能、低延迟
- 可支撑场景：块存储、对象存储、云原生数据库、EC等
- 当前实现了高性能块存储，对接OpenStack和 K8s
网易内部线上无故障稳定运行一年多，线上异常演练
- 已开源
 - [github主页](https://opencurve.github.io/): <https://opencurve.github.io/>
 - [github代码仓库](https://github.com/opencurve/curve): <https://github.com/opencurve/curve>

Tags				
v1.1.0-beta ...				
🕒 on 24 Sep	🔗 5d648c9	📦 zip	📦 tar.gz	📄 Notes ⬇ Downloads
v1.0.0-rc0 ...				
🕒 on 31 Aug	🔗 b1b7845	📦 zip	📦 tar.gz	⬇ Downloads
v1.0.0-beta ...				
🕒 on 17 Aug	🔗 a9c30ec	📦 zip	📦 tar.gz	⬇ Downloads
v0.1.4 ...				
🕒 on 7 Aug	🔗 9e347da	📦 zip	📦 tar.gz	📄 Notes ⬇ Downloads
v0.1.3 ...				
🕒 on 30 Jul	🔗 f6425ad	📦 zip	📦 tar.gz	📄 Notes ⬇ Downloads
v0.1.2 ...				
🕒 on 27 Jul	🔗 af30ce9	📦 zip	📦 tar.gz	📄 Notes ⬇ Downloads



目录

01

背景

为何从0到1开发Curve

02

总体设计

基本架构 | 数据组织形式 | 拓扑 | IO流程

03

系统特性

高性能 | 高可用 | 自治 | 易运维 | 高质量

04

近期规划

Curve的近期规划

- 多个存储软件：SDFS、NEFS、NBS
- 已有的开源软件：Ceph
 - 不能胜任性能、延迟敏感的场景
 - 异常场景抖动较大（比如慢盘场景）
 - 去中心节点设计在集群不均衡的情况下需要人工运维
- 基于通用分布式存储构建上层存储服务



目录

01

背景

为何从0到1开发Curve

02

总体设计

基本架构 | 数据组织形式 | 拓扑 | IO流程

03

系统特性

高性能 | 高可用 | 自治 | 易运维 | 高质量

04

近期规划

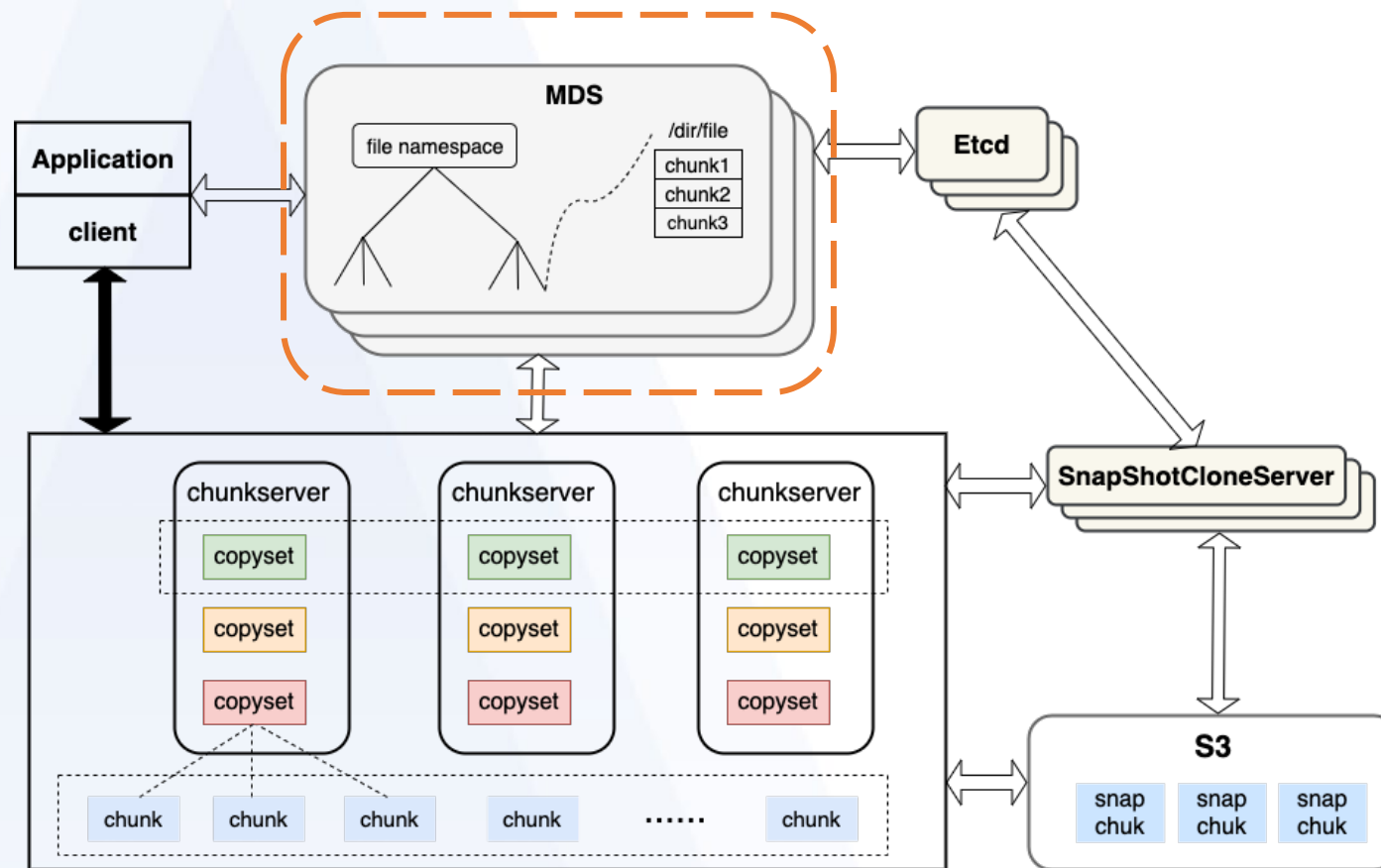
Curve的近期规划

基本架构



- 元数据节点 MDS

管理元数据信息
收集集群状态信息, 自动调度



基本架构



- 元数据节点 MDS

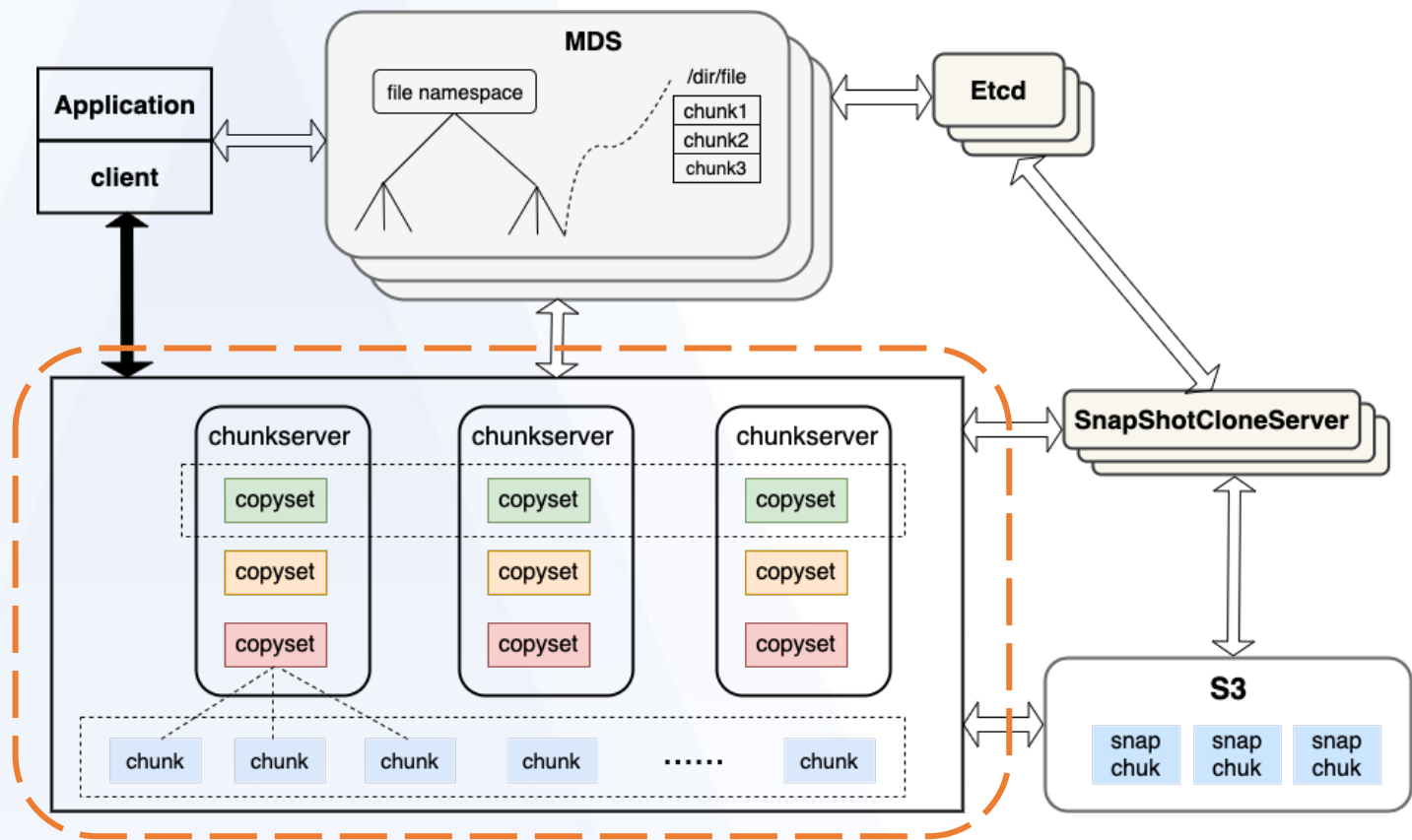
管理元数据信息

收集集群状态信息，自动调度

- 数据节点 Chunkserver

数据存储

数据一致性



基本架构



- 元数据节点 MDS

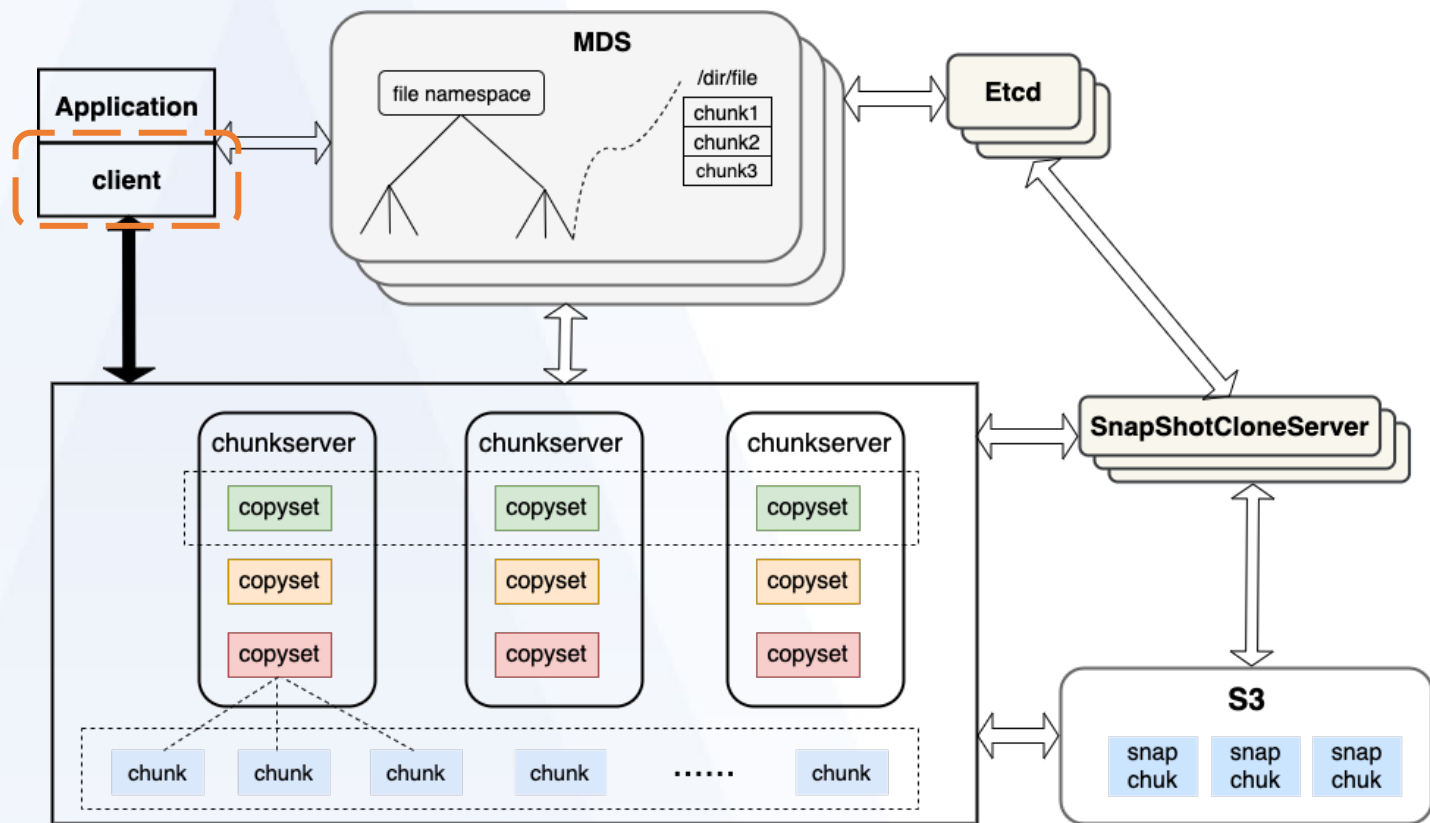
管理元数据信息
收集集群状态信息, 自动调度

- 数据节点 Chunkserver

数据存储
副本一致性

- 客户端 Client

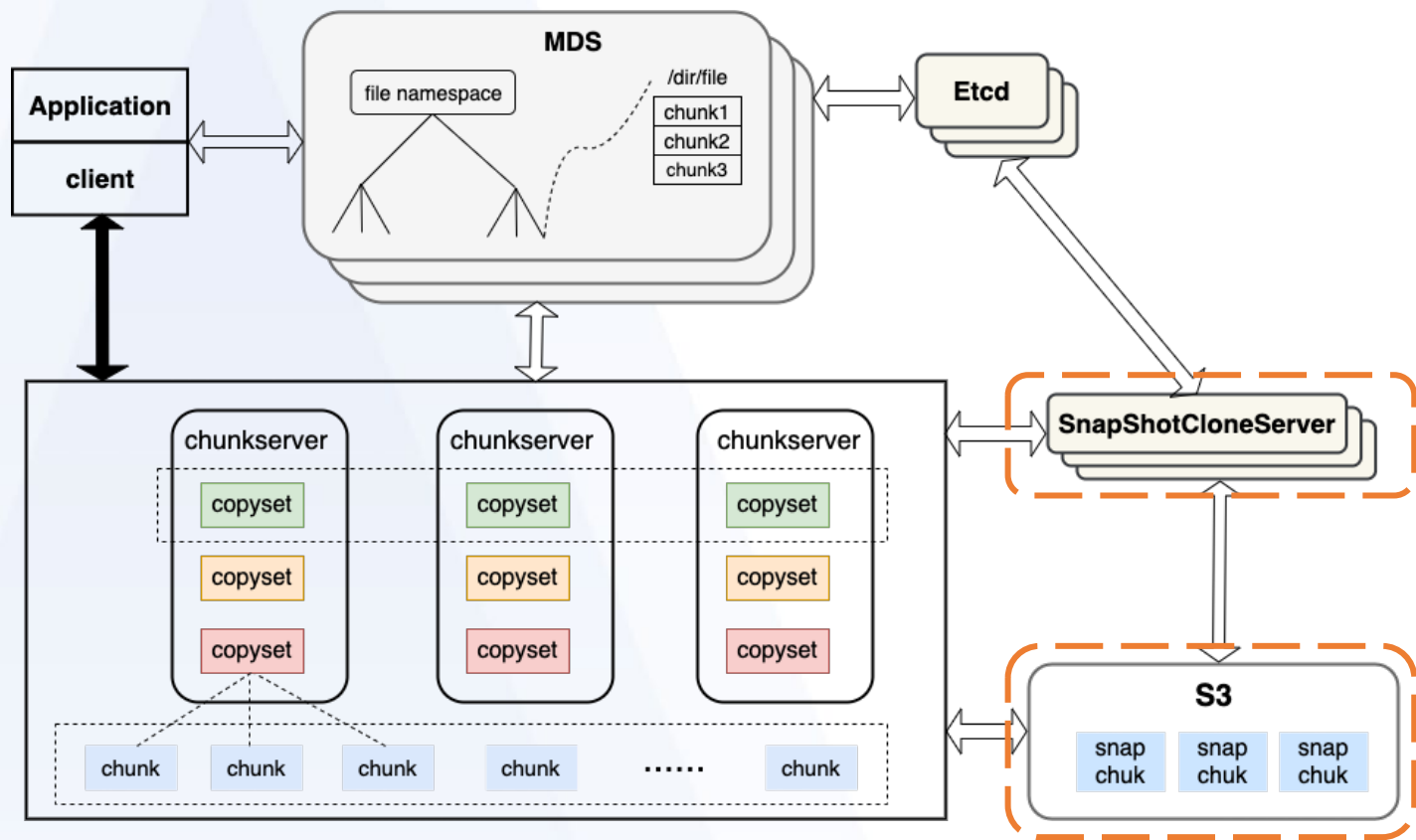
对元数据增删改查
对数据增删改查



基本架构



- 快照克隆服务器
 - 独立于核心服务
 - 储到支持S3接口的对象存储, 不限制数量
 - 异步快照、增量快照
 - 从快照/镜像克隆 (lazy/非lazy)
 - 从快照回滚

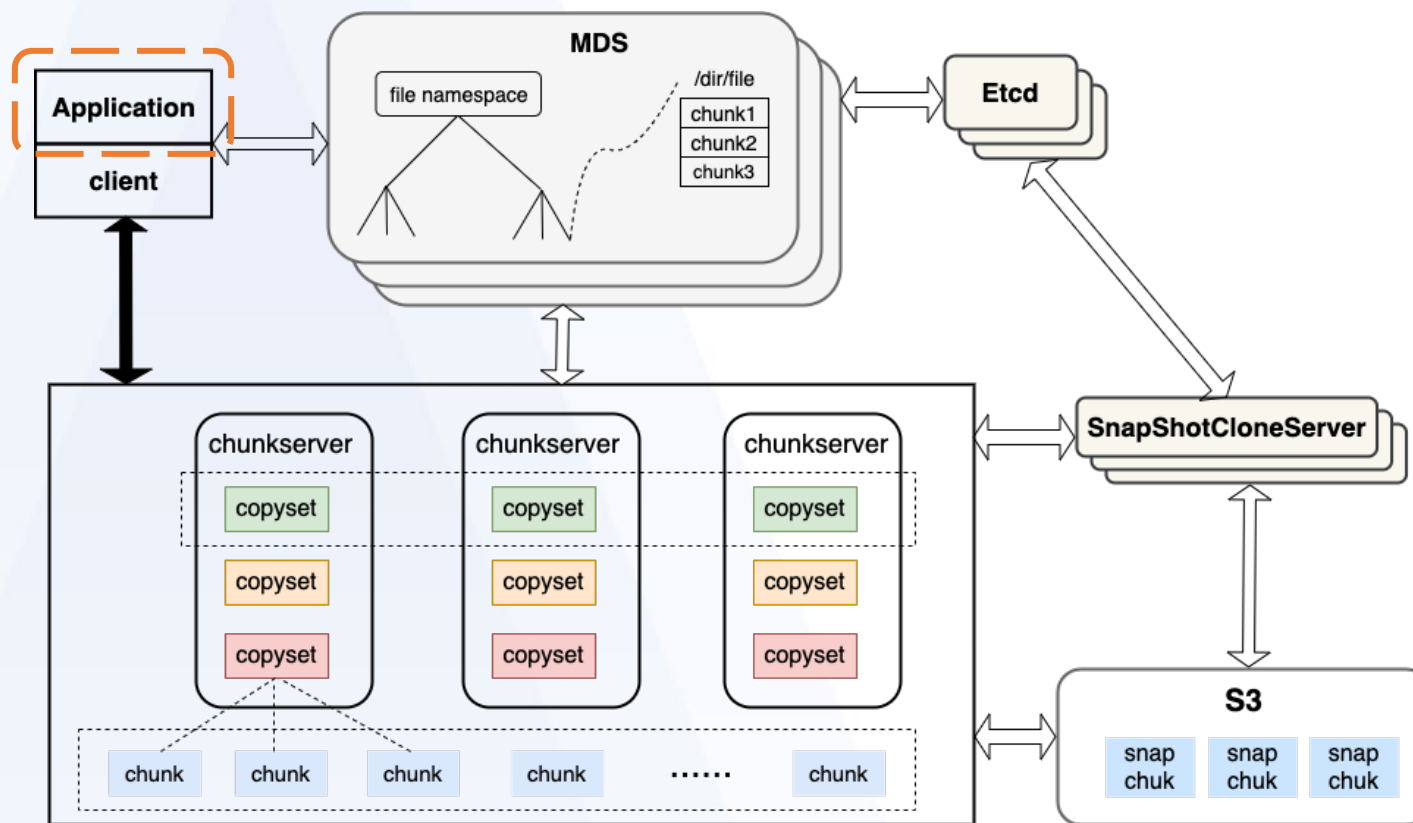


数据组织形式



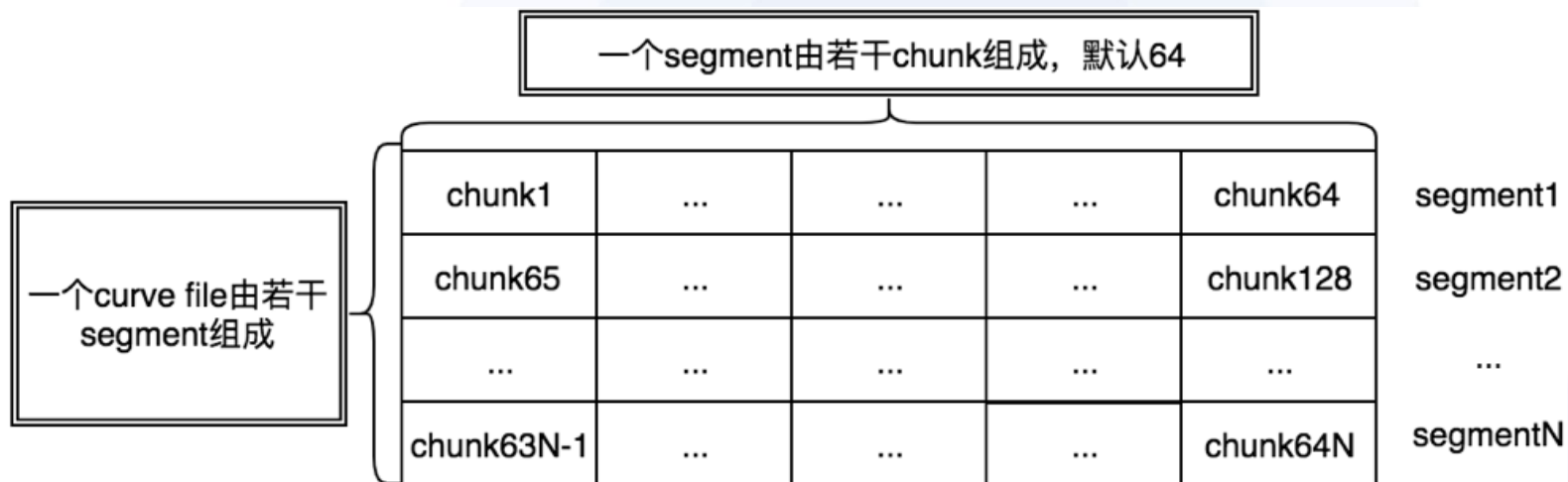
- 底层
 - 可用性 / 可靠性
 - 扩展性 / 负载均衡
 - 向上提供无差别文件流
- Application
 - 块/对象/EC等
 - 感知具体格式

提供不同文件类型支撑不同上层应用



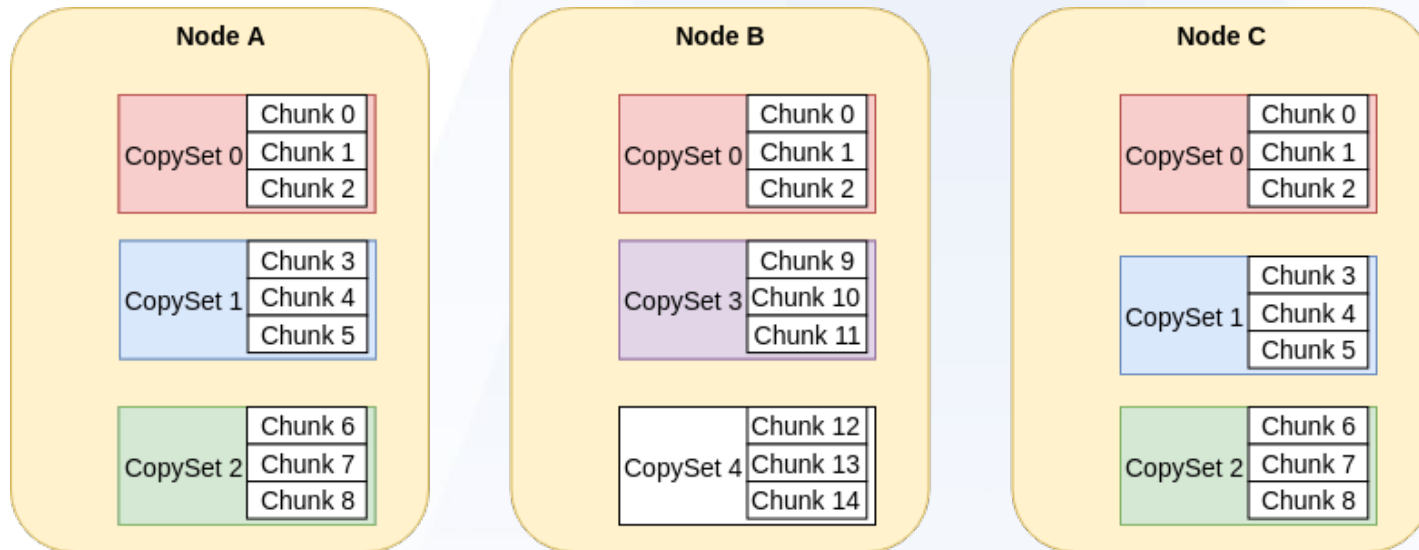
数据组织形式

- PageFile/AppendFile/AppendECFile
- Segment
 - 逻辑概念，空间分配的基本单元（减少元数据数量）
 - 多个连续地址空间chunk（物理文件）的聚合



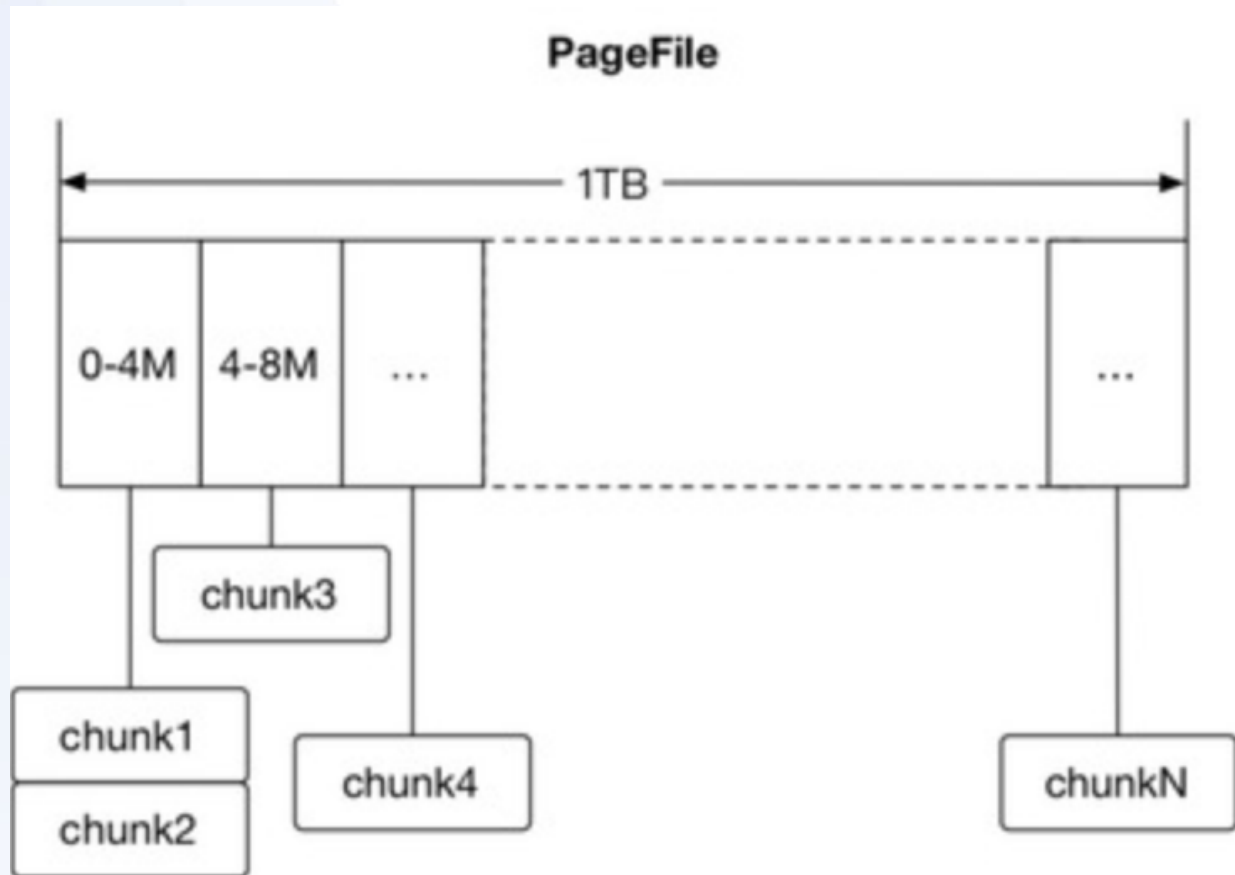
数据组织形式

- CopySet 类似Ceph中的PG [Copysets: Reducing the Frequency of Data Loss in Cloud Storage]
 - 逻辑概念
 - 包含多个chunk
 - 数据放置的基本单元
 - 减少元数据数量
 - 减少复制组数量
 - 提高数据可靠性



数据组织形式

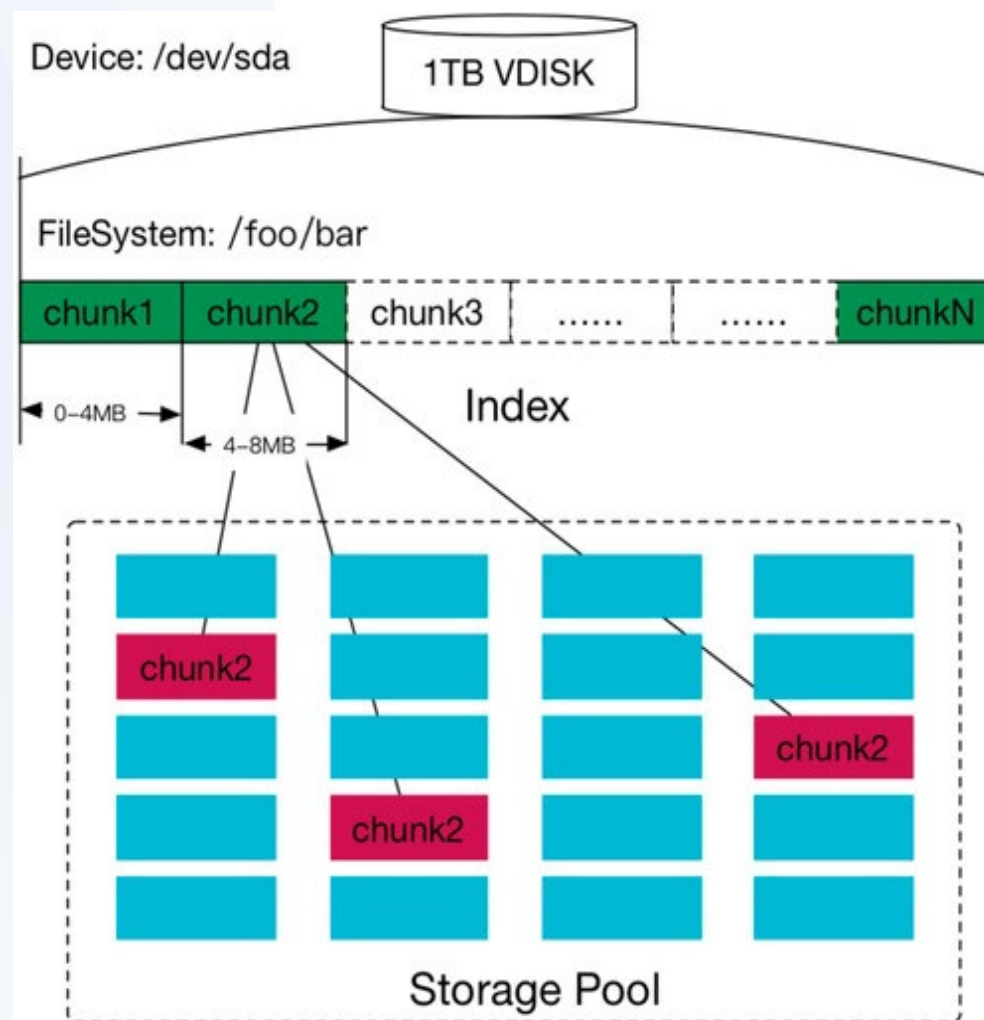
- PageFile
 - 地址空间到—>chunk: 1 : N
 - chunk有先后关系
 - 创建时指定大小, lazy分配chunk
 - 提供4kb随机读写能力



数据组织形式

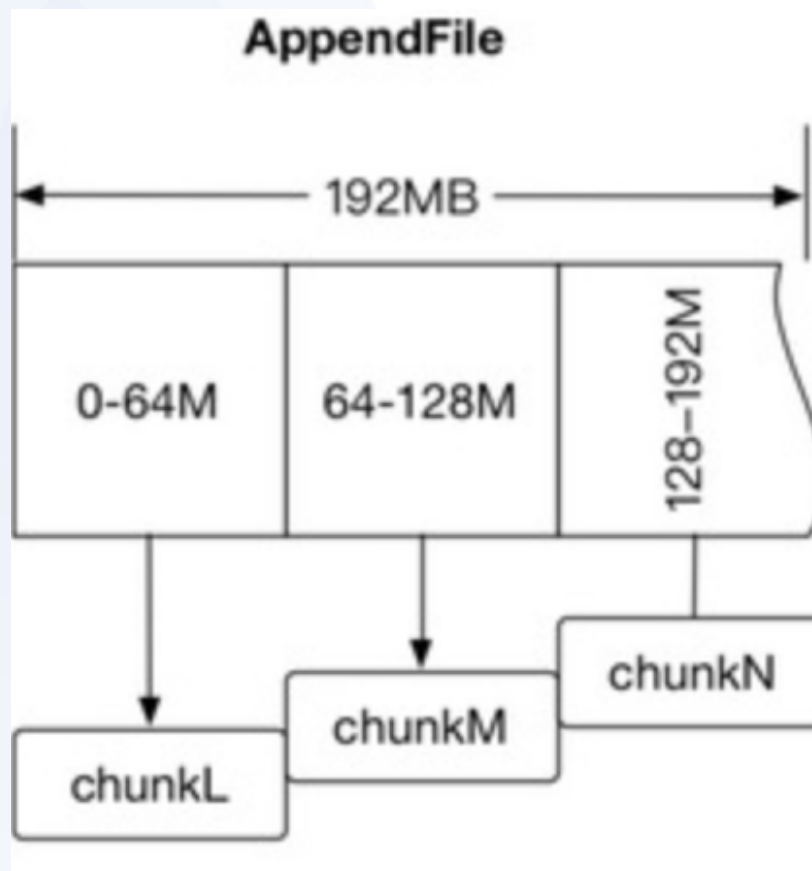
- PageFile

- 地址空间到—>chunk: 1 : N
chunk有先后关系
- 创建时指定大小, lazy分配chunk
- 提供4kb随机读写能力
- 支撑块设备应用场景
块设备层面的快照功能
即为文件层面快照



数据组织形式

- AppendFile
 - 地址空间到—>chunk: 1 : 1
 - 采用append的方式写入

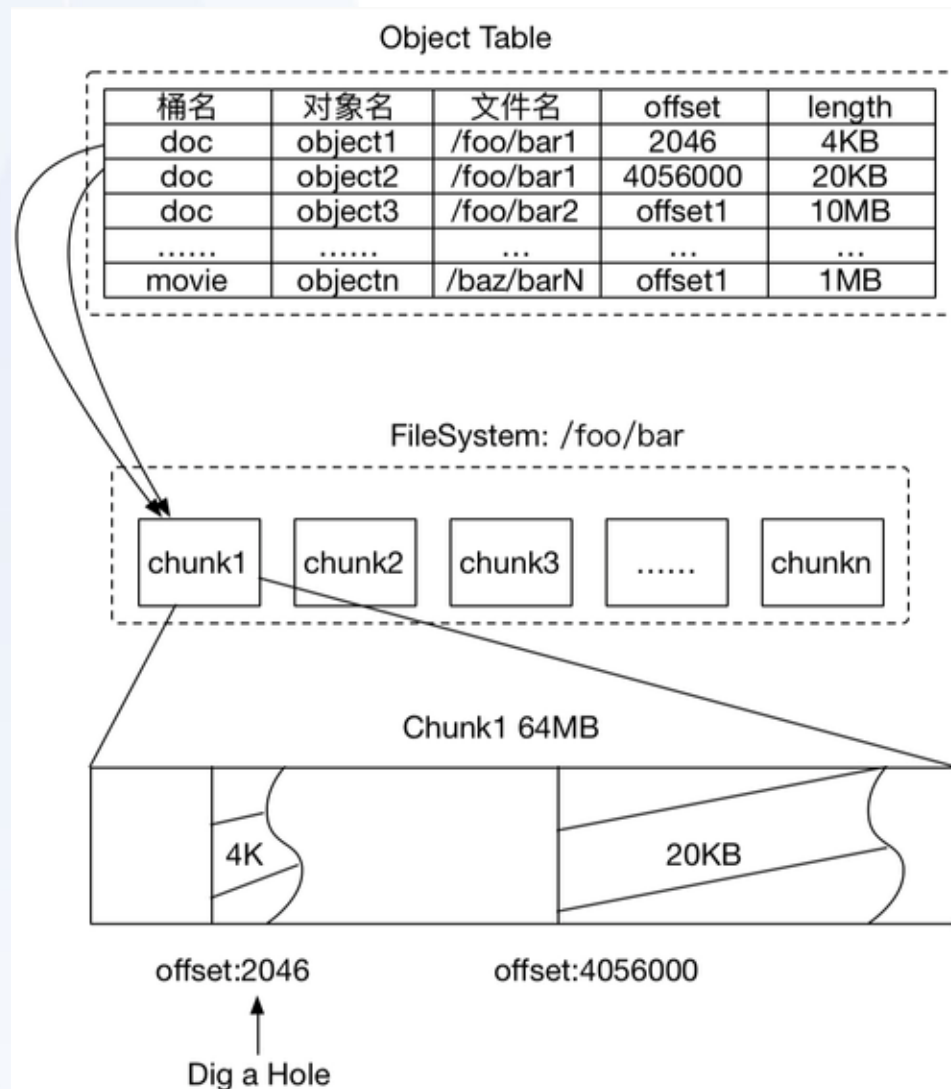


数据组织形式

- AppendFile

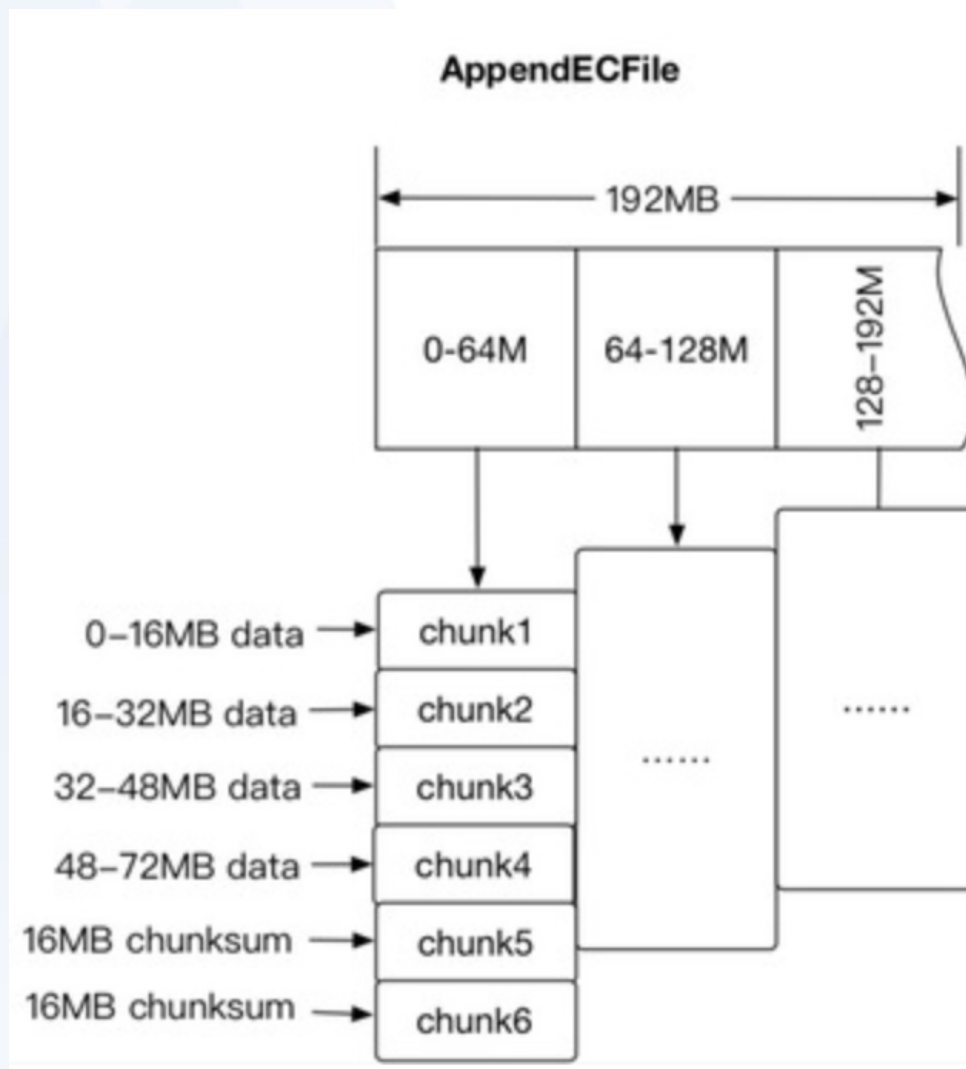
- 地址空间到—>chunk: 1 : 1
- 采用append的方式写入
- 支撑多副本对象存储

通过文件/特殊目录隔离
挖洞即时回收
单独的元信息的存储方案



数据组织形式

- AppendECFile
 - 地址空间到—>chunk: 1 : 1
 - 数据chunk + 校验chunk

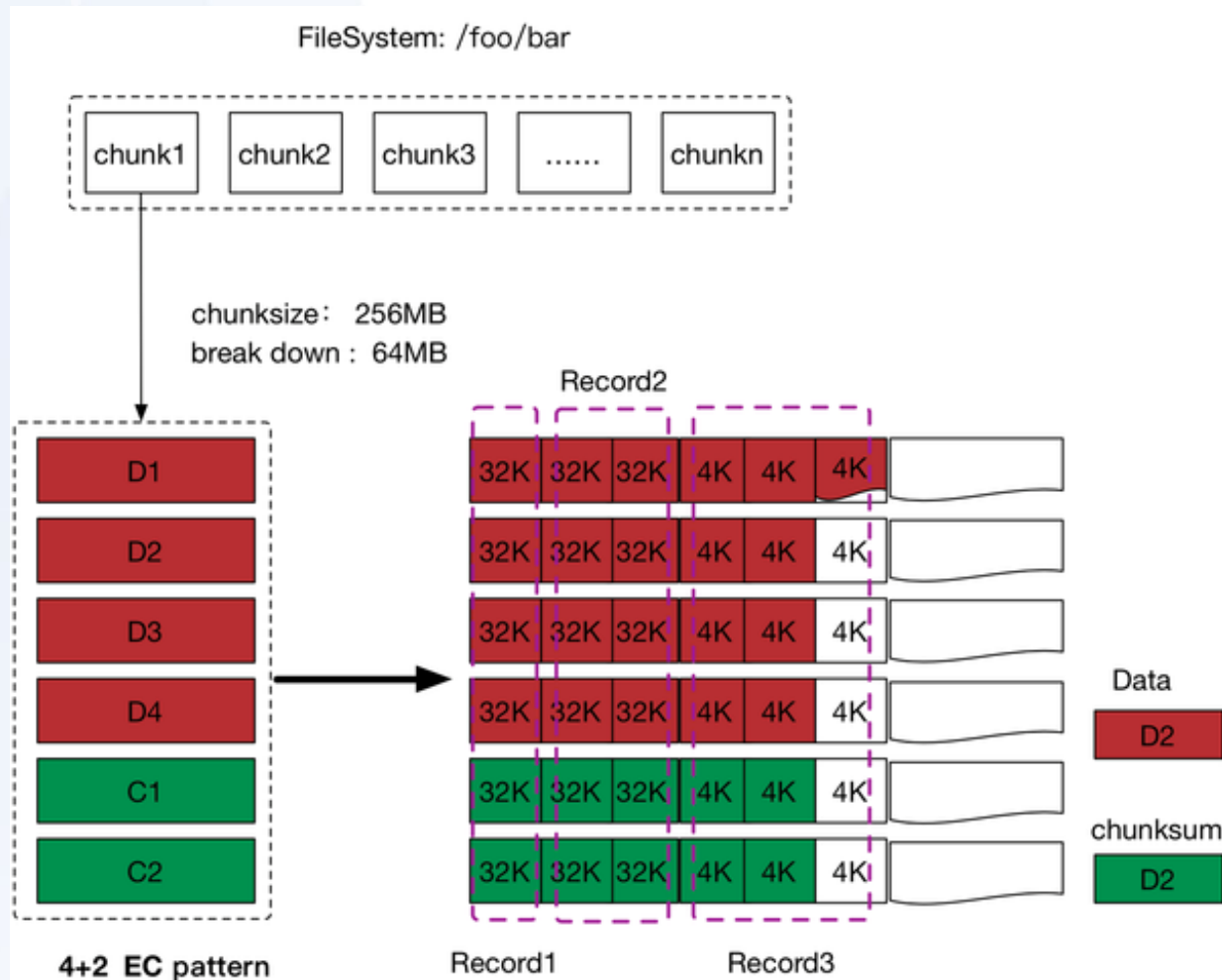


数据组织形式

- AppendECFile

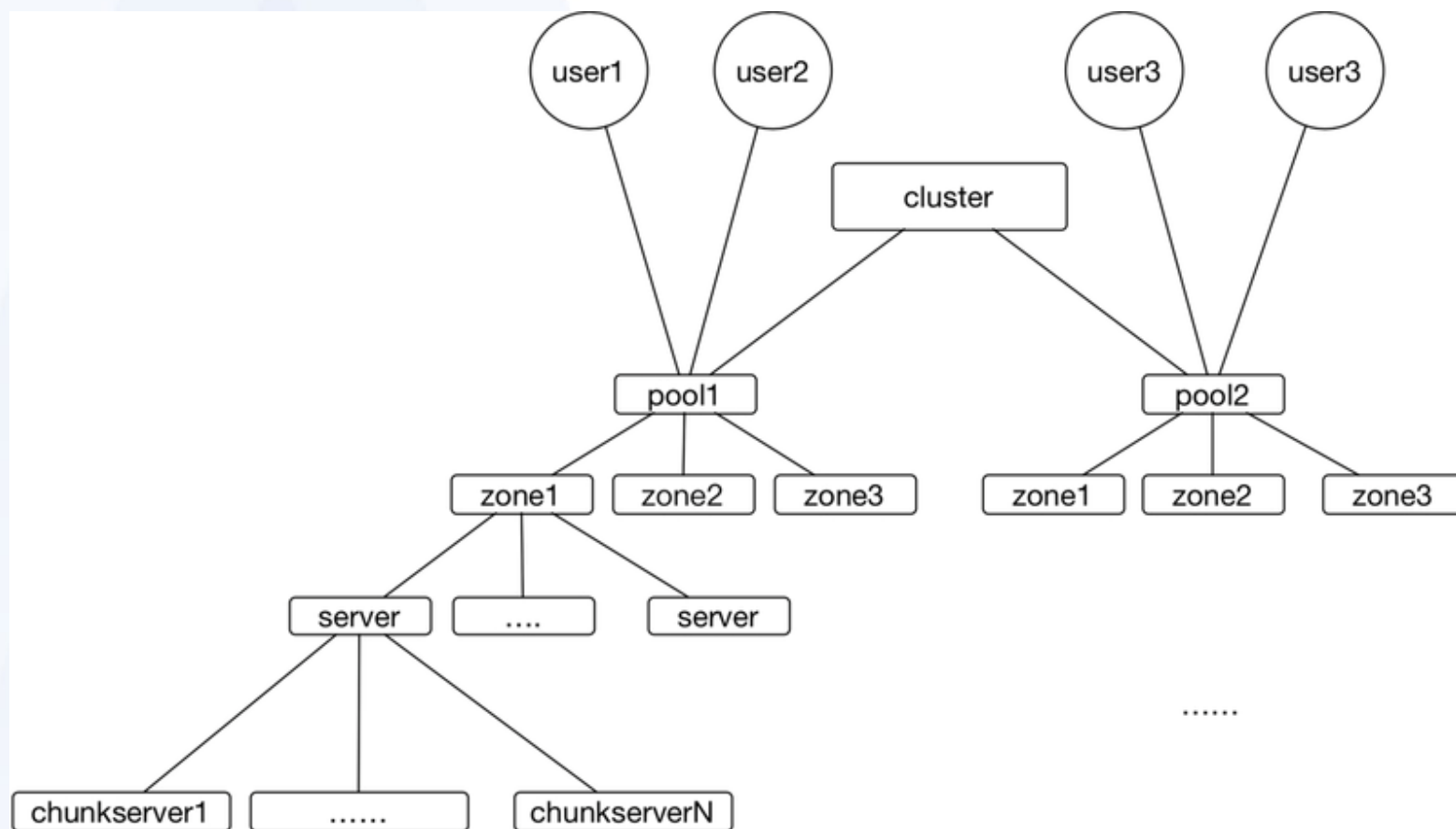
- 地址空间到—>chunk: 1 : 1
- 数据chunk + 校验chunk
- 支撑EC存储场景

多个单副本的 chunk 形成 EC 组
一个对象作为 EC 组的一个满条带
挖洞即时空间回收



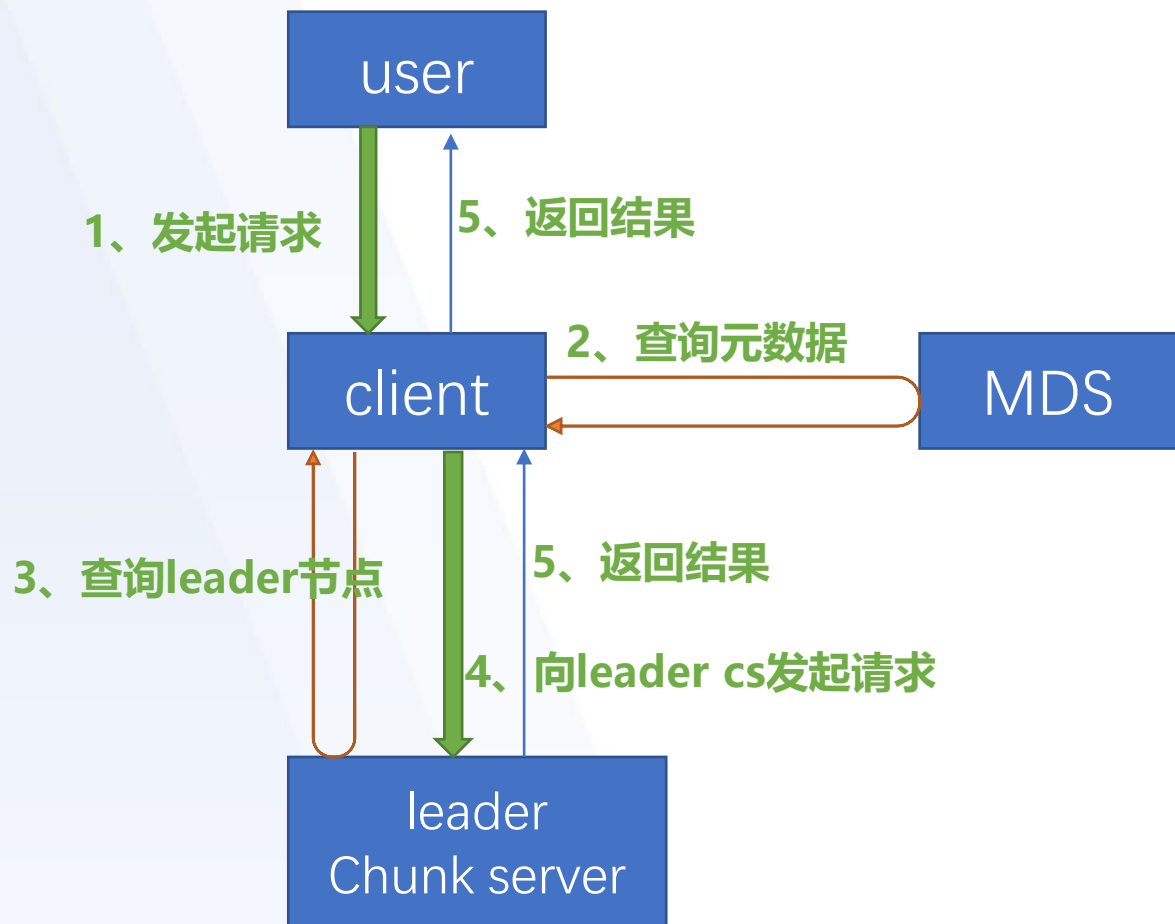
拓扑

- 管理和组织机器
- 软件单元: chunkserver
- 物理机: server
- 故障域: zone
- 物理池: pool



IO流程

1. 用户发起请求;
2. Client 向 mds 查询请求的元数据,
并缓存到本地, 请求转换为对 chunk 的请求
3. Client 向 chunkserver 查询 chunk 所在的
copyset的leader Chunkserver节点;
4. Client 向 leader 发送读写请求,
Chunkserver 完成后通知client;
5. Client通知用户请求完成。





目录

01

背景

为何从0到1开发Curve

02

总体设计

基本架构 | 数据组织形式 | 拓扑 | IO流程

03

系统特性

高性能 | 高可用 | 自治 | 易运维 | 高质量

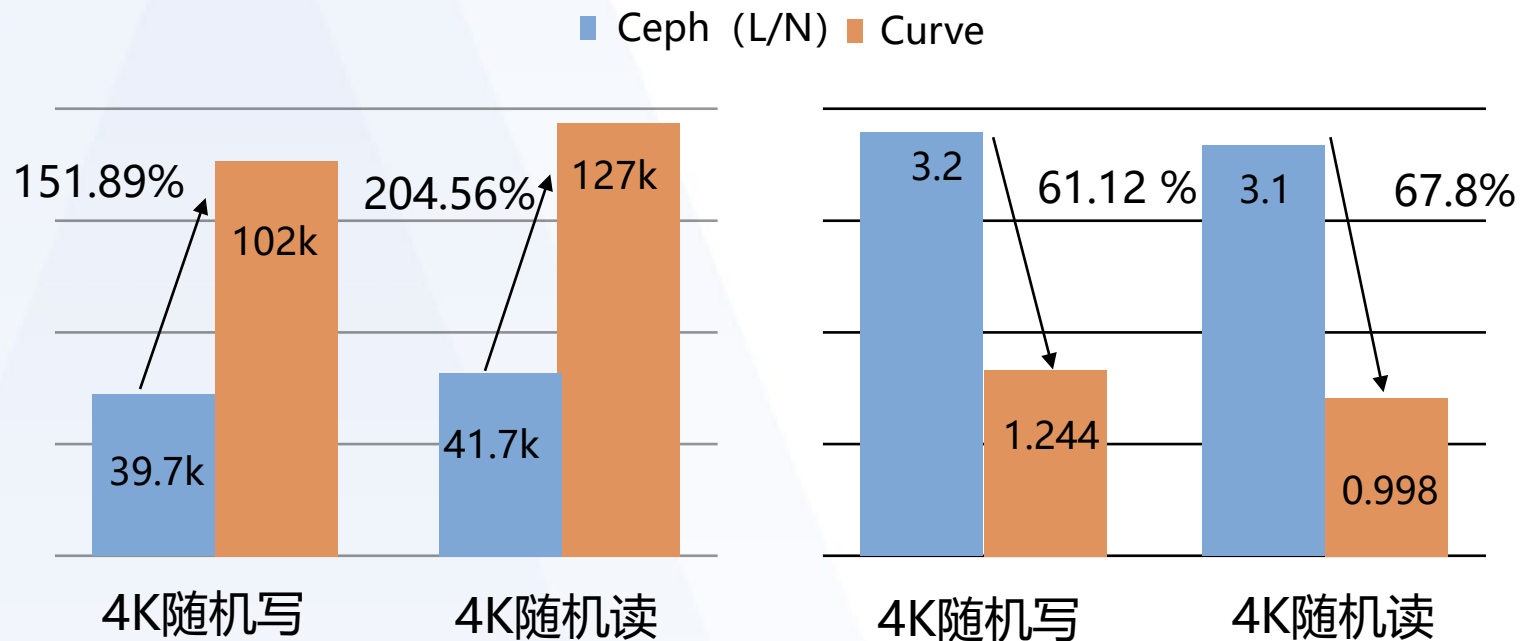
04

近期规划

Curve的近期规划

单卷4K随机读写IOPS

单卷4K随机读写平均延迟(ms)



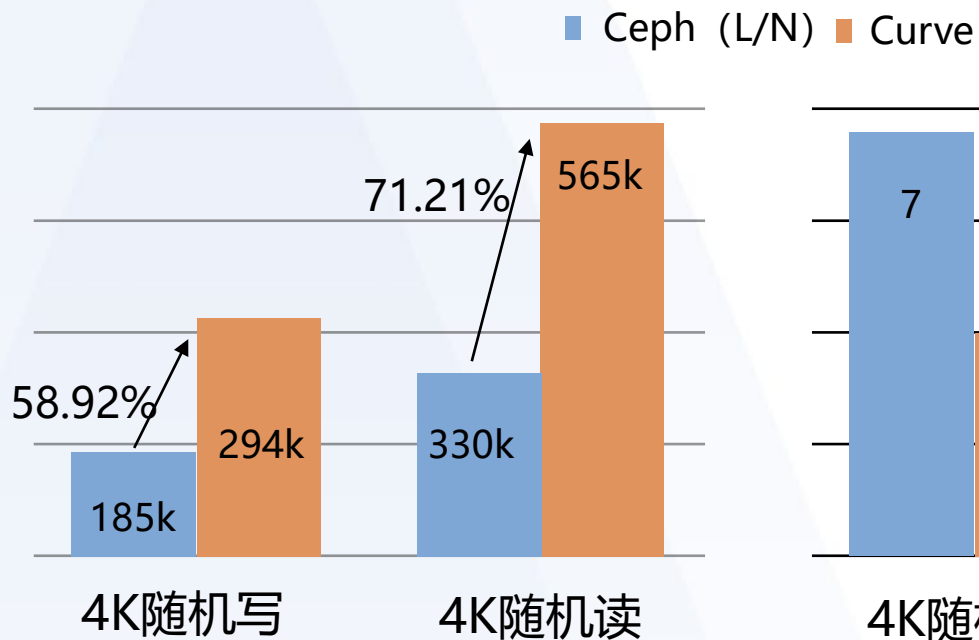
测试环境：6台服务器*20块SATA SSD，E5-2660 v4，256G，3副本场景

高性能

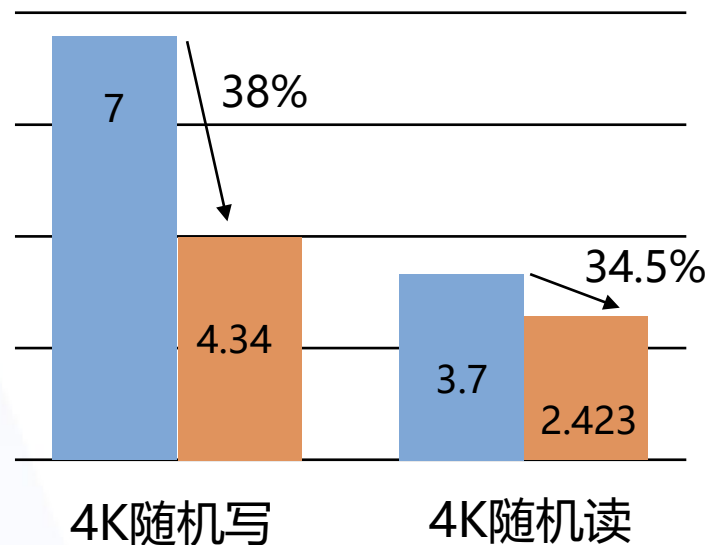


- quorum机制: raft
 - 轻量级快照
- io路径上的优化
 - filepool落盘零放大
 - 轻量级线性一致性读
 - io路径上用户空间零拷贝

10卷4K随机读写IOPS



10卷4K随机读写平均延迟(ms)

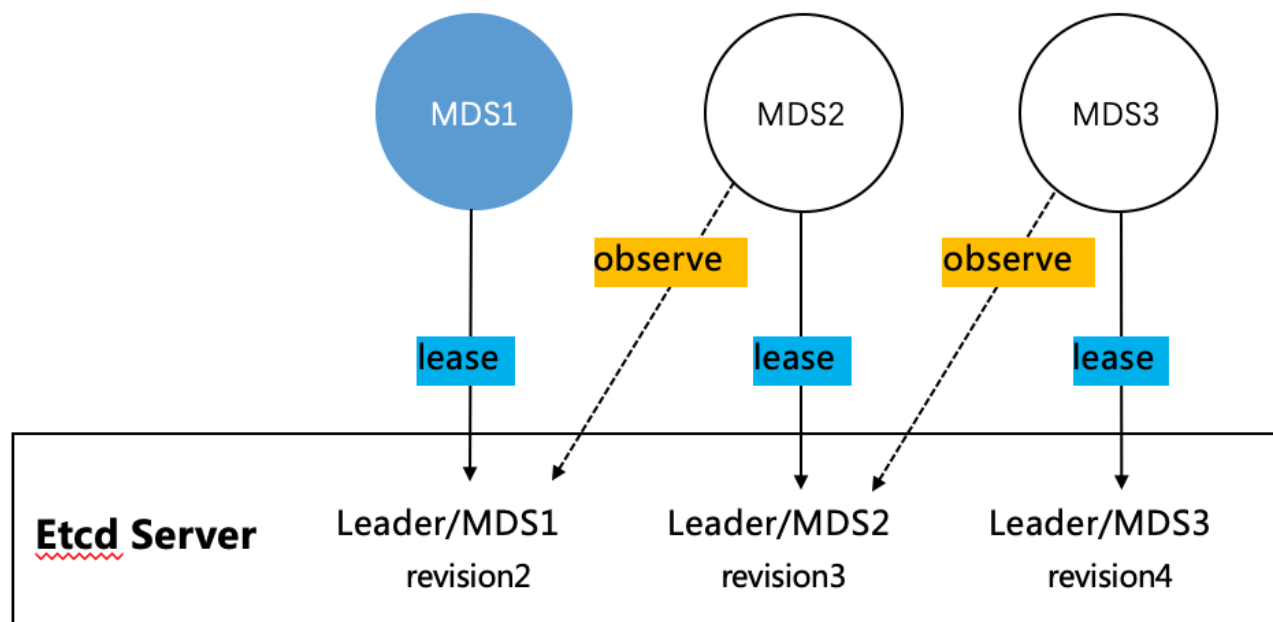


测试环境: 6台服务器*20块SATA SSD, E5-2660 v4, 256G, 3副本场景

高可用

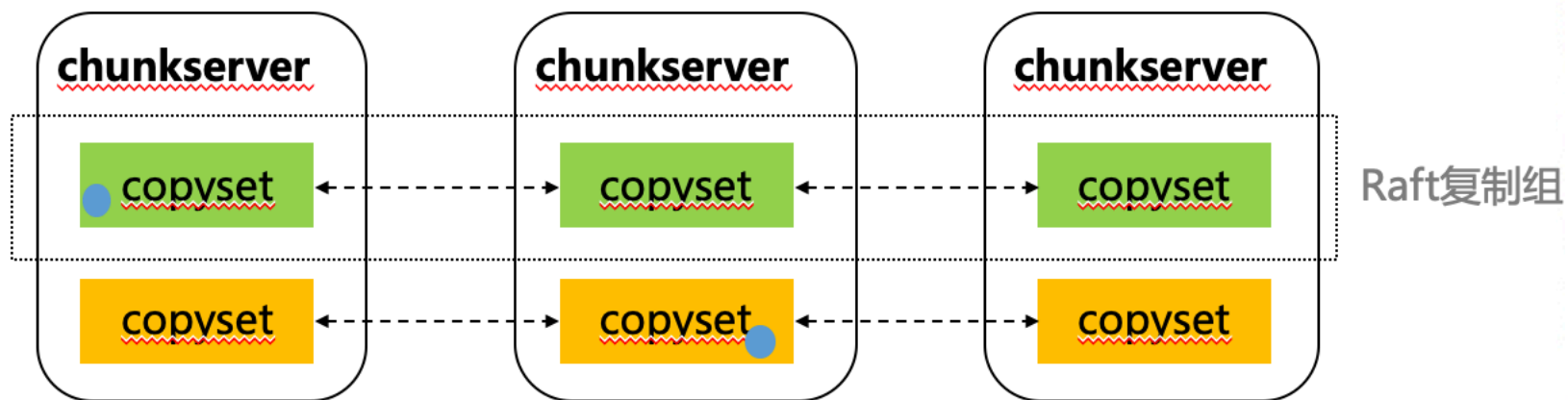
核心组件支持多实例部署，允许部分实例异常

MDS、Snapshotcloneserver 通过 etcd 选主，实现高可用



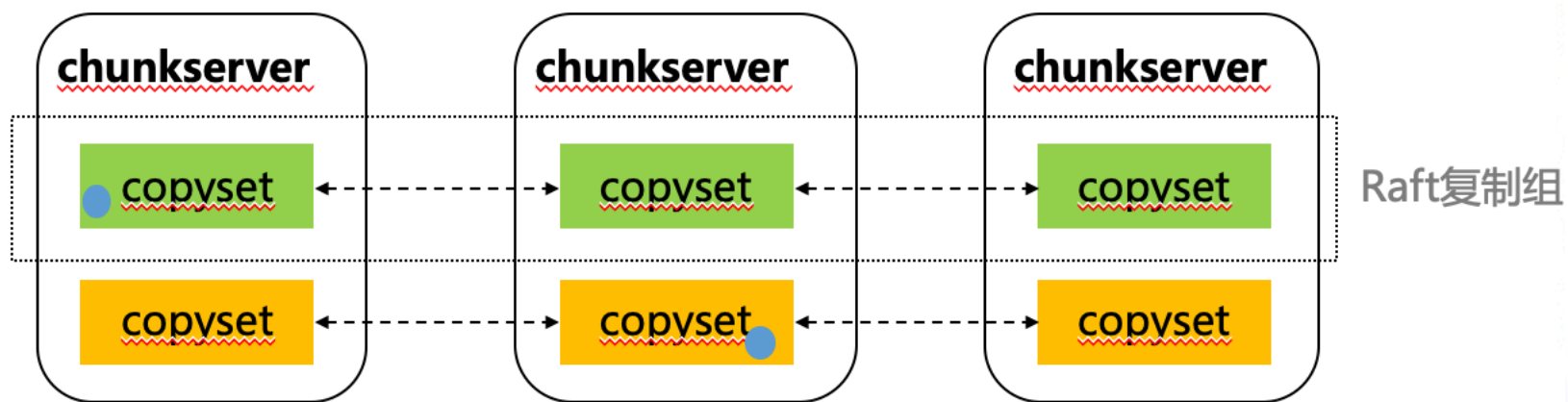
高可用

chunkserver 使用raft, $2N + 1$ 个副本允许 N 副本异常



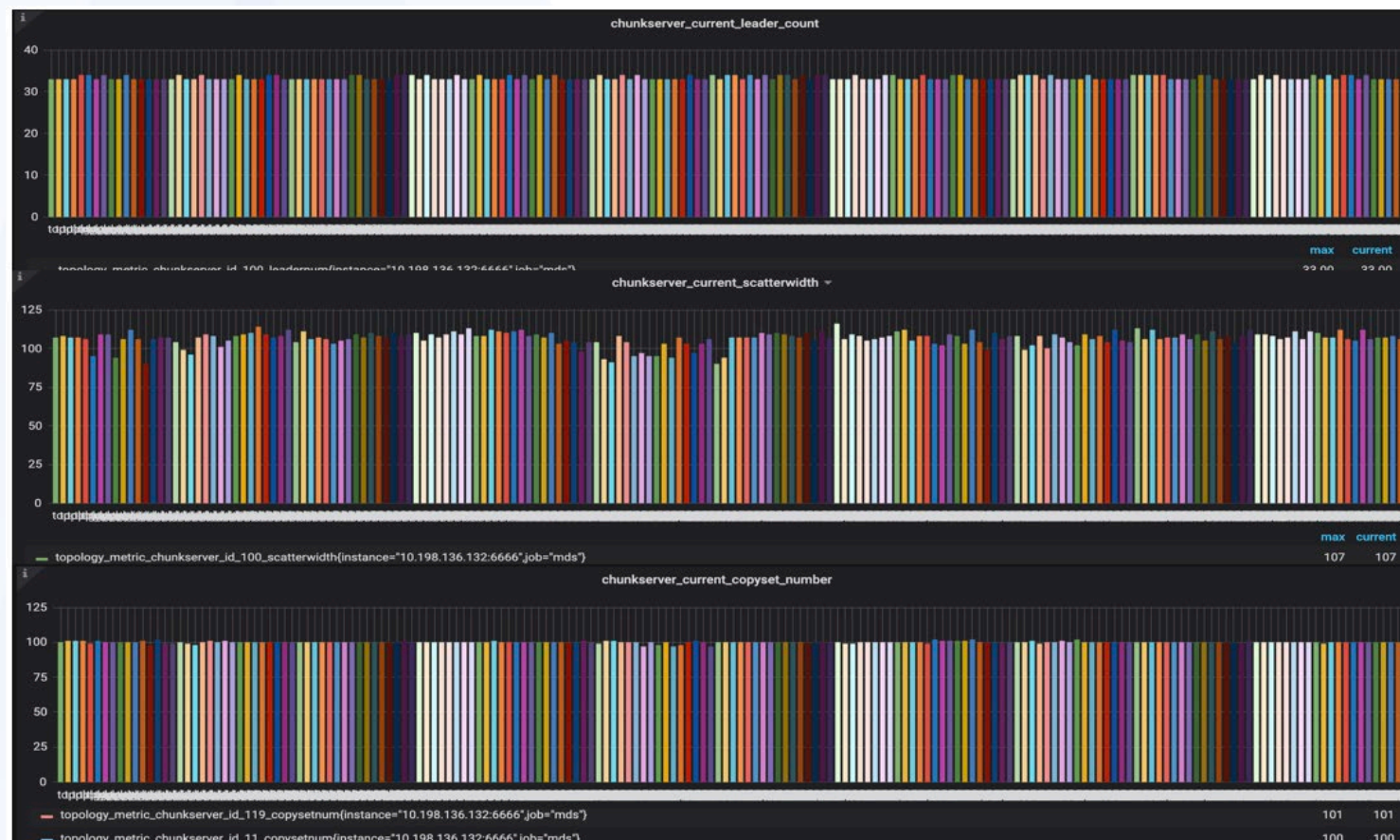
自治

- 自动故障恢复
 - 多对多，恢复时间短
 - 精确的流量控制，对io几乎无影响

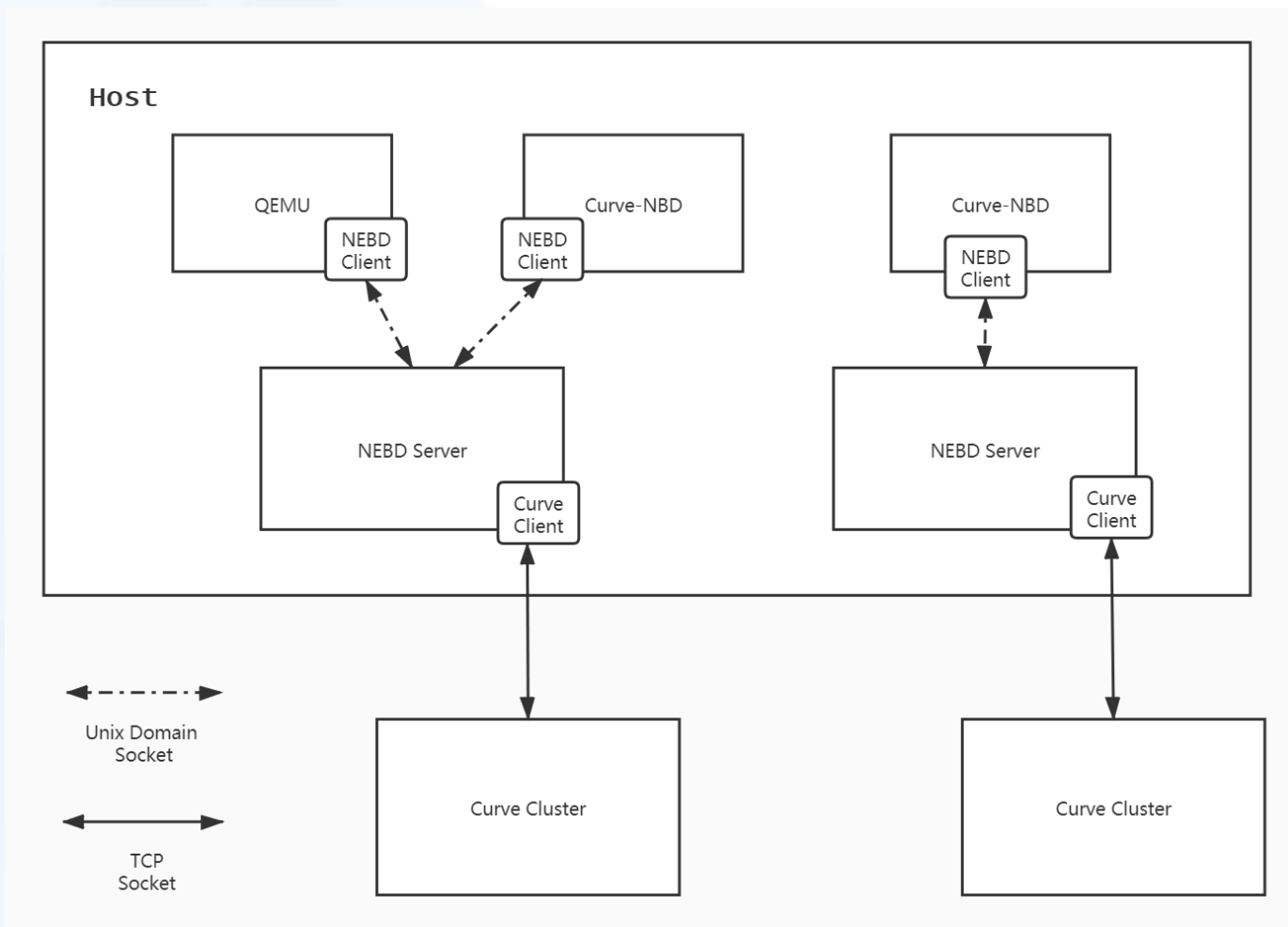


自治

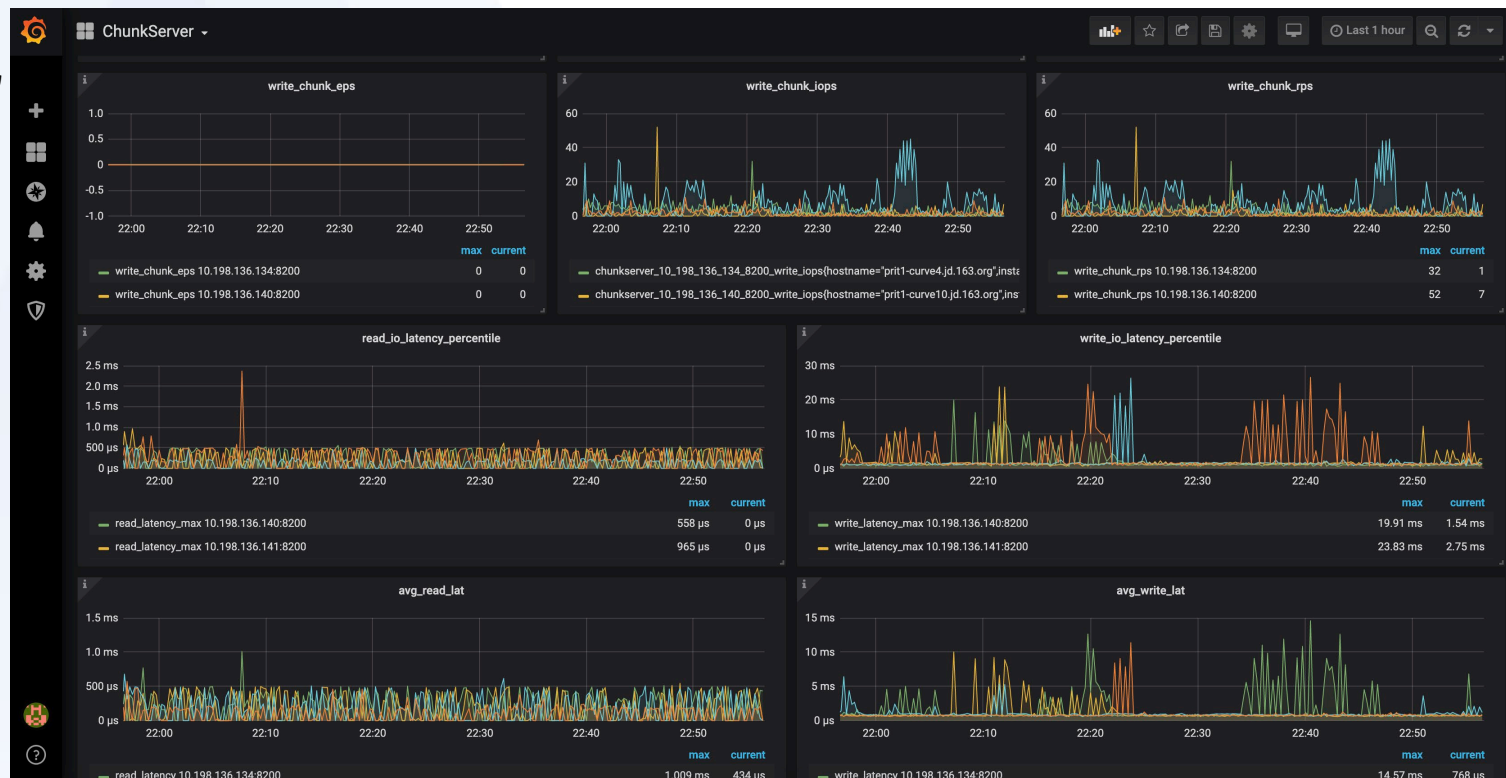
- 集群负载和资源均衡
 - leader
copyset
scatter-width
 - 无需人工干预
 - 对io影响几乎无影响



- 升级秒级影响
 - 客户端采用CS架构
 - NEBD Client: 对接上层业务
 - NEBD Server: 接受请求调用Curve Client处理
 - 升级只需重启Server秒级影响



- 丰富的metric体系
 - prometheus + grafana 可视化
 - 每日报表
 - 丰富的数据定位问题



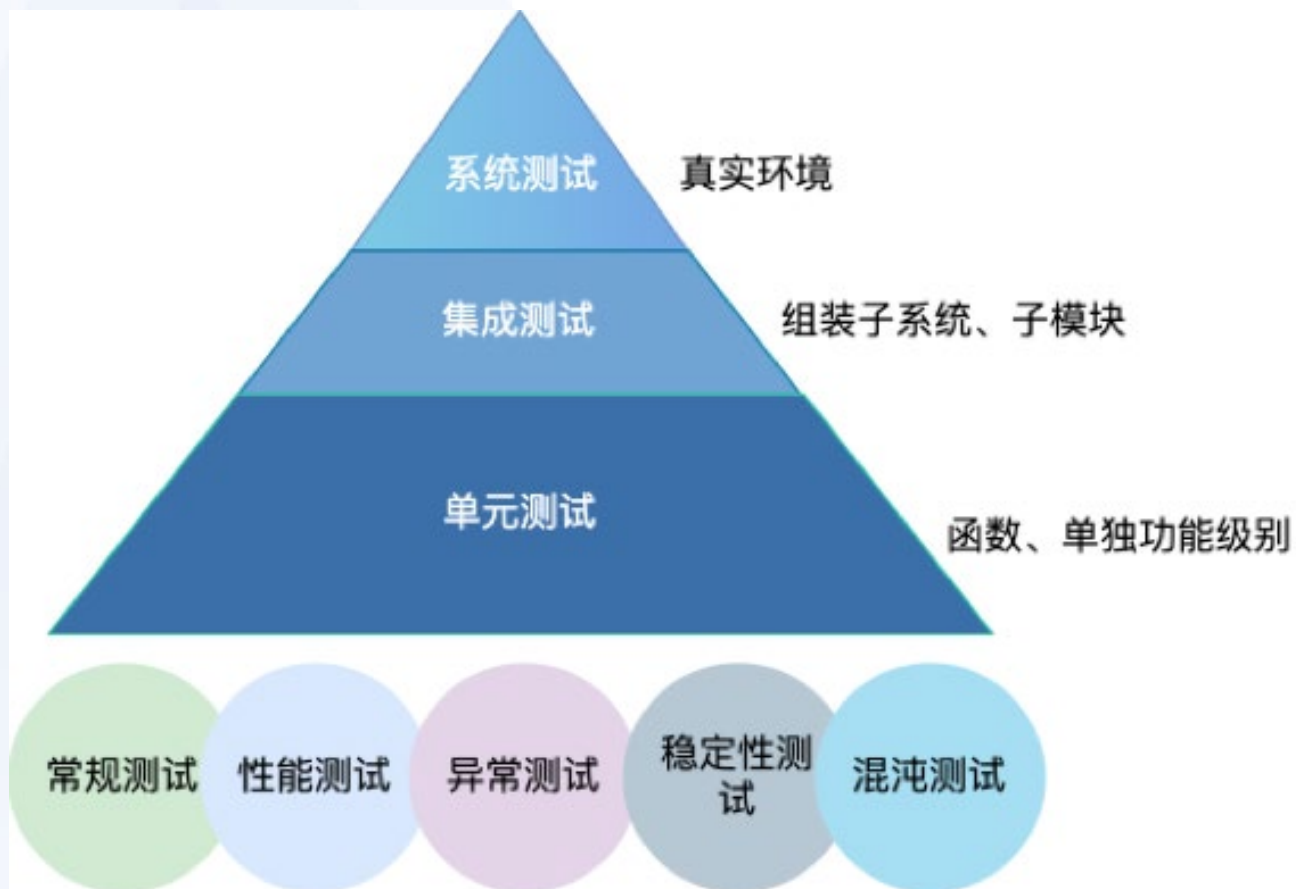
- 丰富的metric体系
 - prometheus + grafana 可视化
 - 每日报表
 - 丰富的数据定位问题
- 集群状态查询工具
 - curve_ops_tool
- 自动化部署工具
 - 一键部署，一键升级

```
Usage: curve_ops_tool [Command] [OPTIONS...]
COMMANDS:
space : show curve all disk type space, include total space and used space
status : show the total status of the cluster
chunkserver-status : show the chunkserver online status
mds-status : show the mds status
client-status : show the client status
etcd-status : show the etcd status
snapshot-clone-status : show the snapshot clone server status
copysets-status : check the health state of all copysets
chunkserver-list : show curve chunkserver-list, list all chunkserver information
get : show the file info and the actual space of file
list : list the file info of files in the directory
seginfo : list the segments info of the file
delete : delete the file, to force delete, should specify the --forcedelete=true
clean-recycle : clean the RecycleBin
create : create file, file length unit is GB
chunk-location : query the location of the chunk corresponding to the offset
check-consistency : check the consistency of three copies
remove-peer : remove the peer from the copyset
transfer-leader : transfer the leader of the copyset to the peer
reset-peer : reset the configuration of copyset, only reset to one peer is supported
check-chunkserver : check the health state of the chunkserver
check-copyset : check the health state of one copyset
check-server : check the health state of the server
check-operator : check the operators
rapid-leader-schedule: rapid leader schedule in cluster in logicalpool
```

高质量



- 良好的模块化和抽象设计
- 完善的测试体系
 - 单元测试
行覆盖80%+, 分支覆盖70%+
 - 集成测试
Given When Then 方法
完备的测试用例集
 - 自动化异常测试
41个异常用例
 - 自动化大压力随机故障注入
20轮随机故障注入





目录

01

背景

为何从0到1开发Curve

02

总体设计

基本架构 | 数据组织形式 | 拓扑 | IO流程

03

系统特性

高性能 | 高可用 | 自治 | 易运维 | 高质量

04

近期规划

Curve的近期规划

近期规划



- 性能优化
 - 满足数据库性能要求
 - 大io吞吐优化
 - muti raft 性能优化
- 开源
 - 系列技术分享
 - 参与开发人员线上议题讨论会

欢迎大家参与CURVE项目！



- [github主页](https://opencurve.github.io/): <https://opencurve.github.io/>
- [github代码仓库](https://github.com/opencurve/curve): <https://github.com/opencurve/curve>
- [系列讲座](https://space.bilibili.com/700847536/channel/detail?cid=153949): <https://space.bilibili.com/700847536/channel/detail?cid=153949>

THANK YOU

D I G I T A L S A I L



扫码即可关注