# 网易新一代对象存储引擎

孙建良

# 关于我

- 孙建良
- 网易
  - 图片处理系统
  - 小文件缓存系统
  - 广域网上传加速系统
  - 新一代对象存储引擎

**blog: work-jlsun.github.io**

# Object Storage vs HDFS

# HDFS

- Summary

✓ unstructured data in arbitary formats

✓ Block, usually 64MB.

✓ Blocks are replicated.

✓ Write once (append allowed)

✓ (Often) collocated with compute capacity.

# Object Store

- what is an object store?

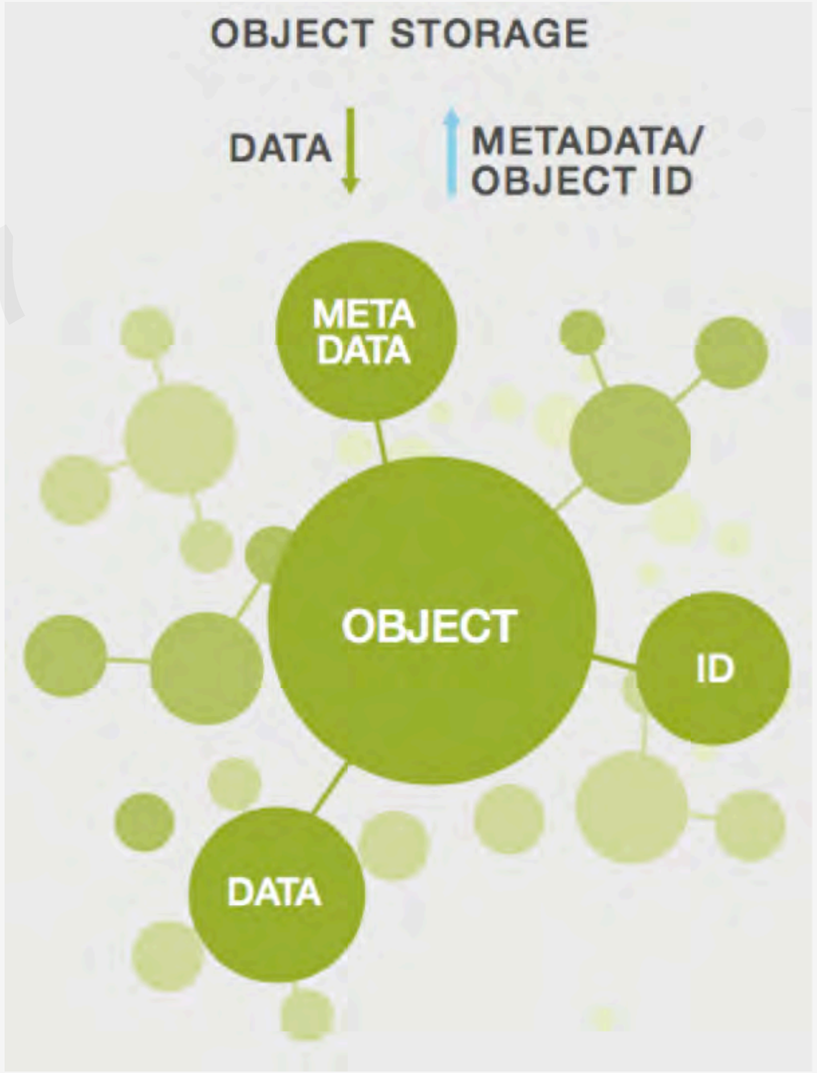
  ✓ Key

  ✓ Value

  ✓ Attribute

  ✓ Bucket

  ✓ RestFul HTTP: https://bucket.nos.netease.com/doc.txt 、SDK



# Object Store

- Good things about object stores

✓ (effectively) infinitely scalable – EB and beyond.

✓ Various security models – data is safe.

✓ Low cost, long term storage solution.

# Object Store

- Object Storage is Not a File System

  ✓ Write once – no append in place

  ✓ Usually eventual consistent

  ✓ No Real DIR

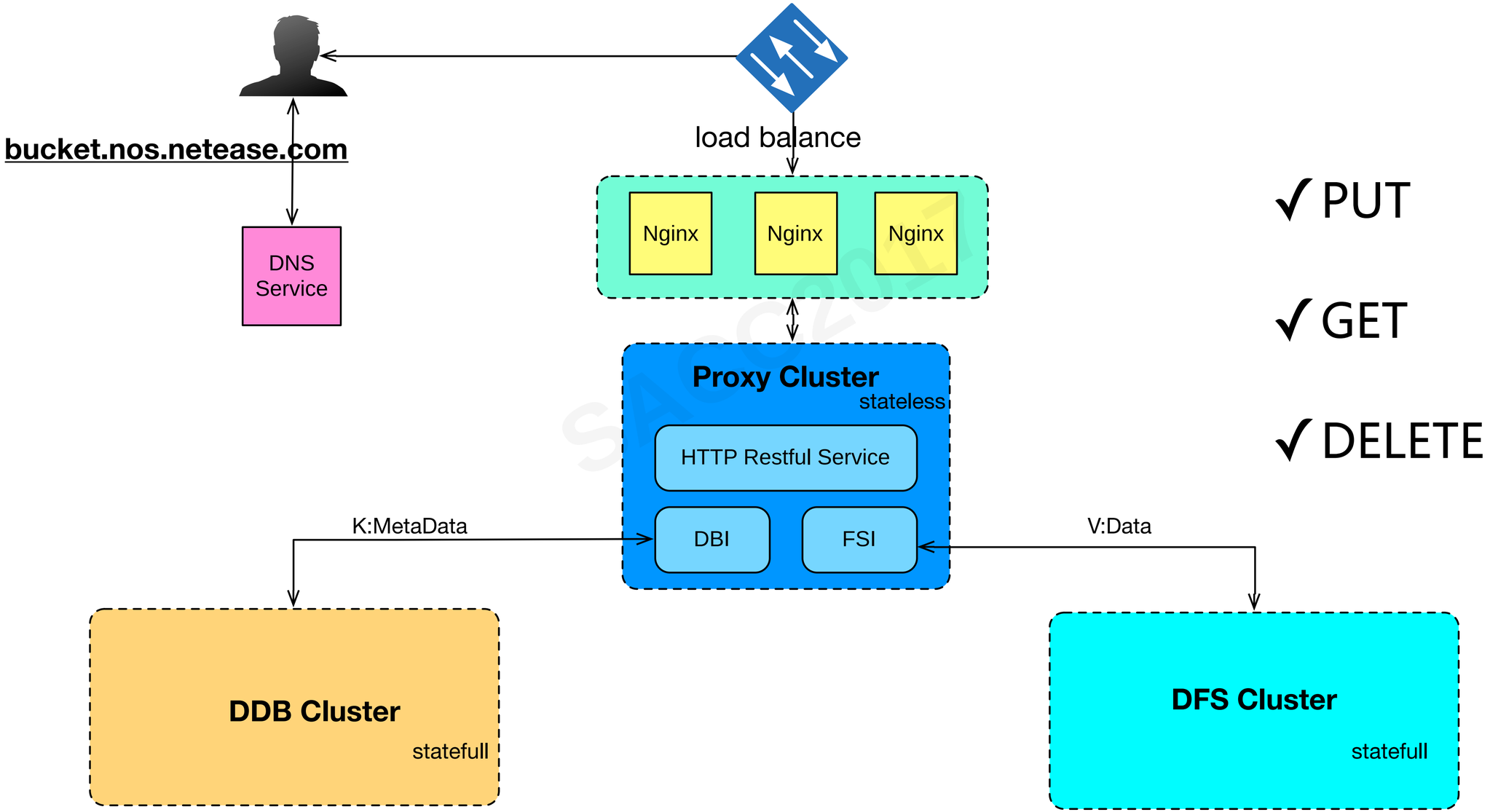
# Outline

BasicArch

背景

NEFS

# 对象存储基础架构




✓ PUT

✓ GET

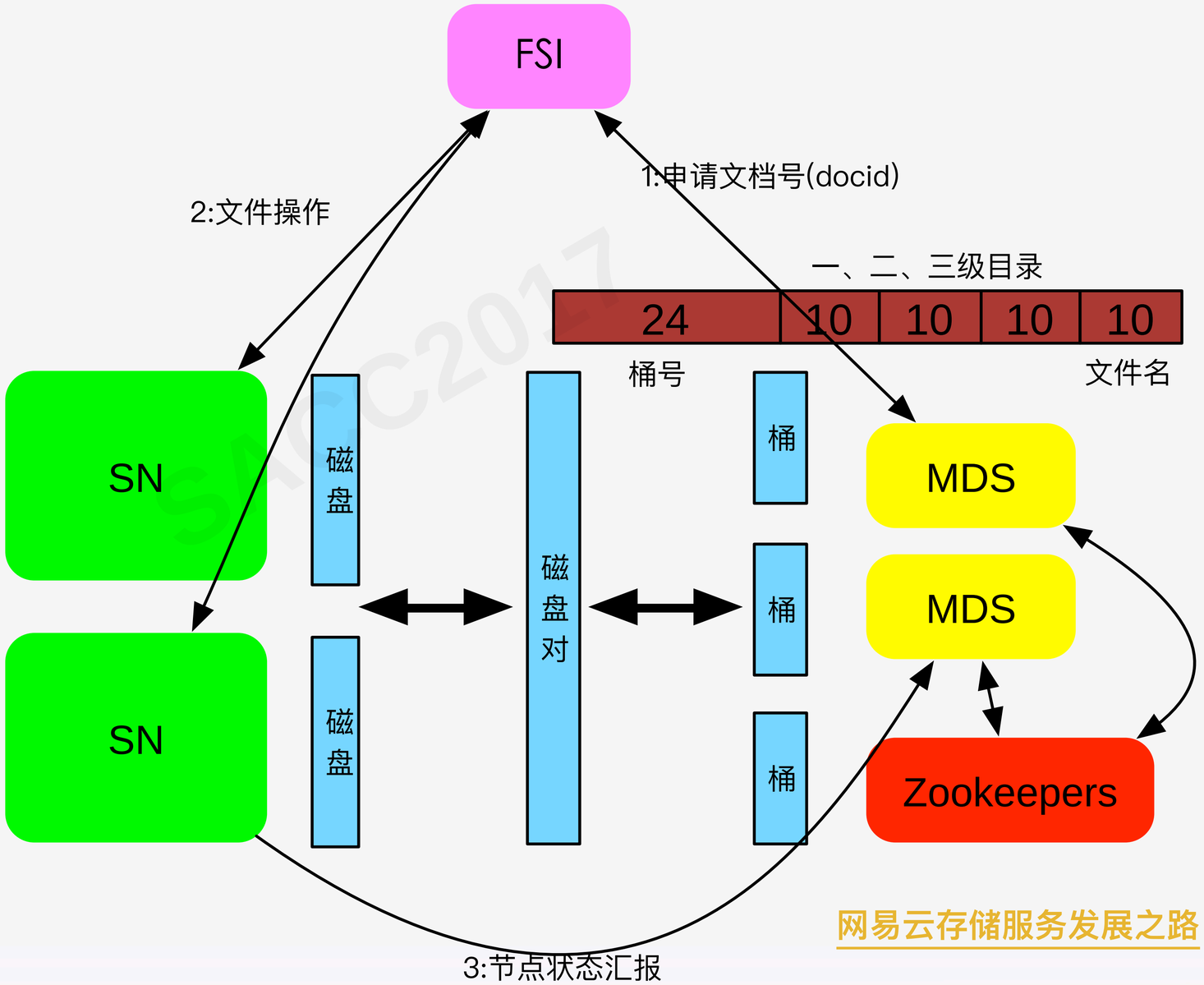
✓ DELETE

# 背景-DFS

- 分布式框架

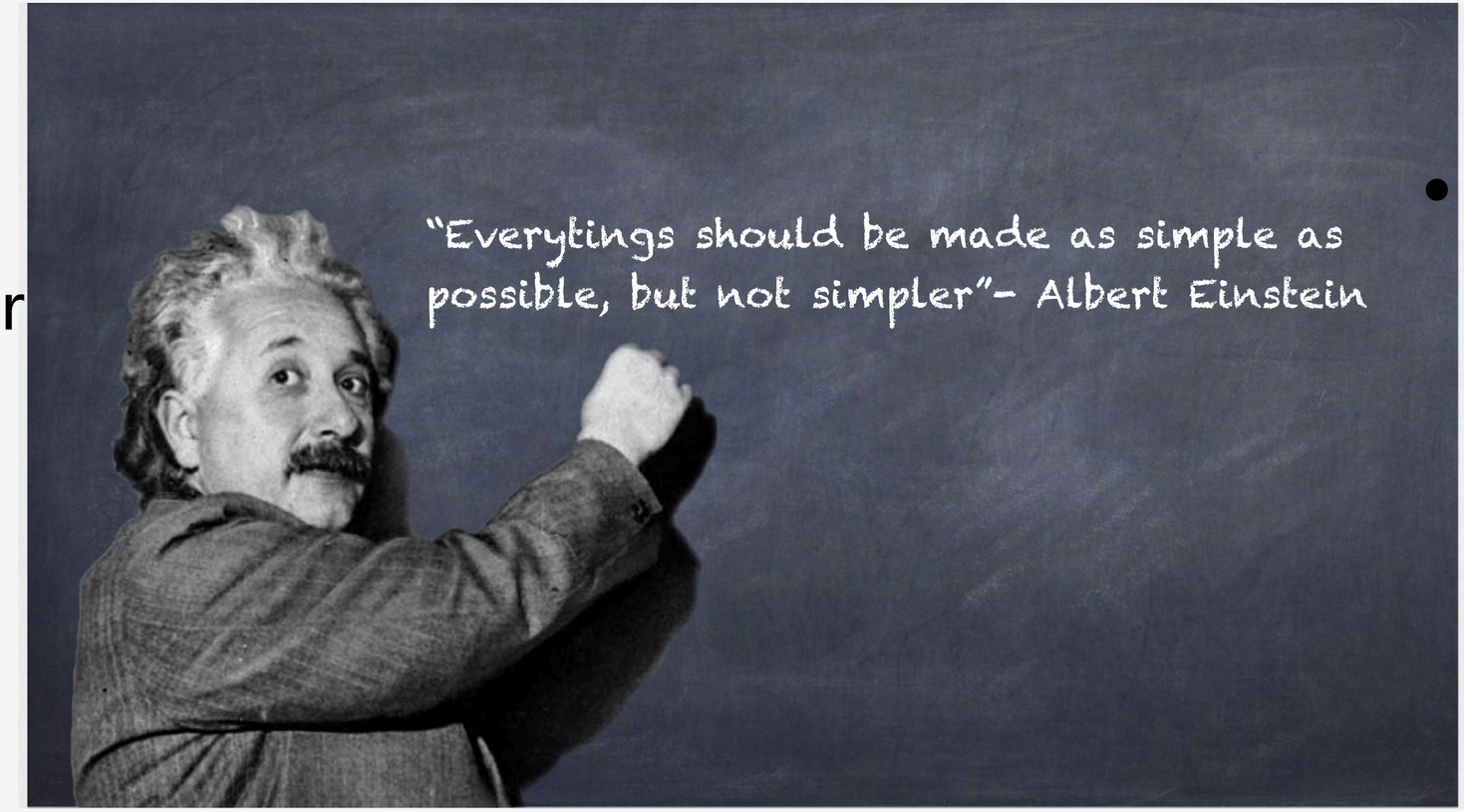
✓副本组织形式

✓数据写入



# 背景

- 缺点
  - it is simpler
    - ✓性能
    - ✓可靠性
    - ✓成本




- 优点
  - ✓简单、简单、简单
  - ✓复制组、一致性、引

# Design Goals

- ✓ Capacity：100PB+
- ✓ WorkLoad：适应大小文件
- ✓ Durability：8个9、11个9
- ✓ Availability： 机架感知、组件高可用、减小依赖
- ✓ Scale Easy: 灵活、不影响性能、支持Rebalance
- ✓ MultiTalent：多租户
- ✓ Simple：Keep it Simple

# NEFS

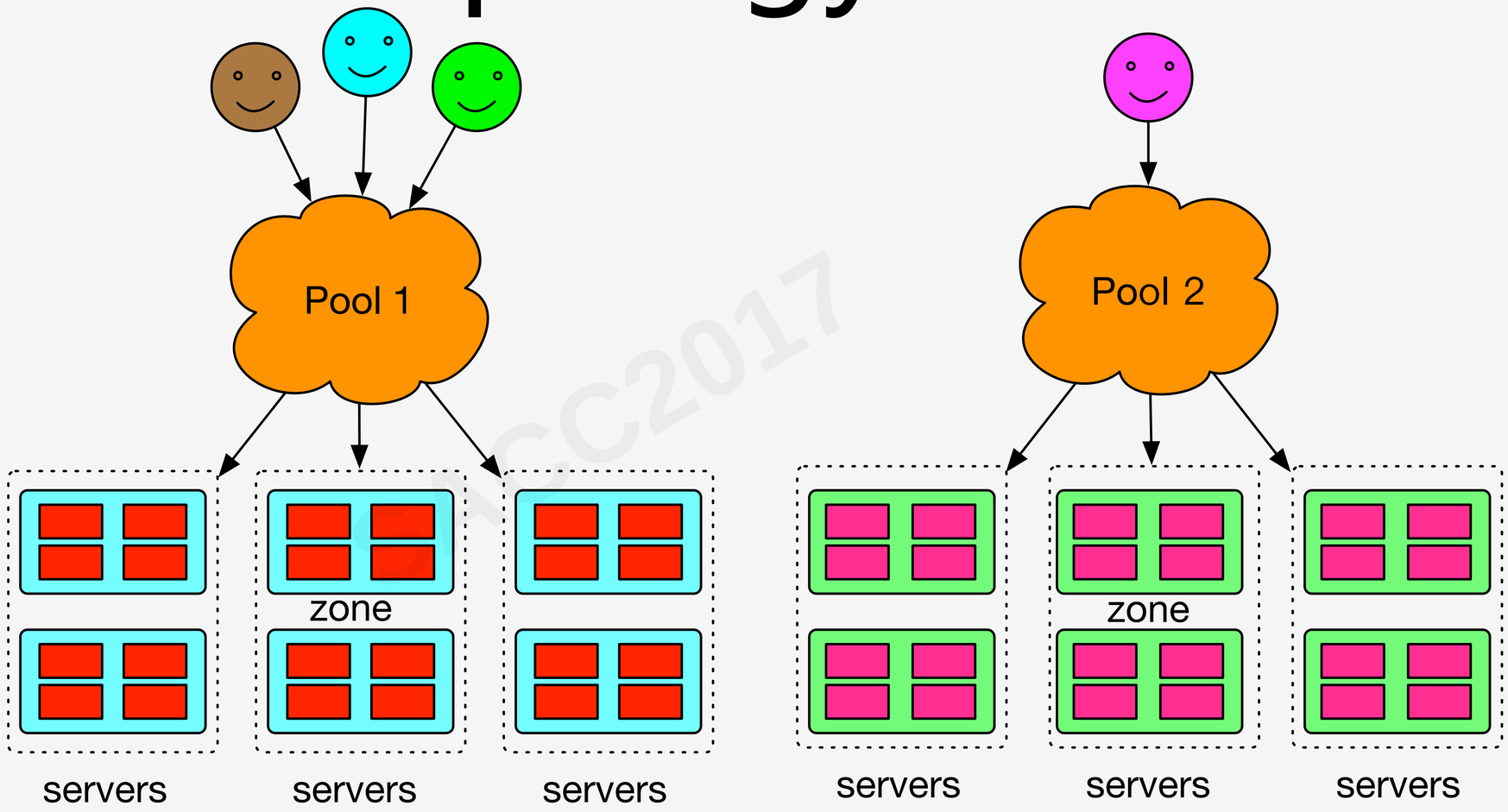
# Overview

- Netease File System（NEFS）
  - ✓ Key-Value Blob Storage
  - ✓ Key：FID（16 Byte）（8+8）
  - ✓ Value : Blob (an arbitrary-sized byte Chunk)

- Interface
  - ✓ PutFile :: User、Blob -> FID
  - ✓ GetFile::FID ->Blob
  - ✓ DeleteFile:FID->bool
  - ✓ GetFileInfo:FID->FileStatus

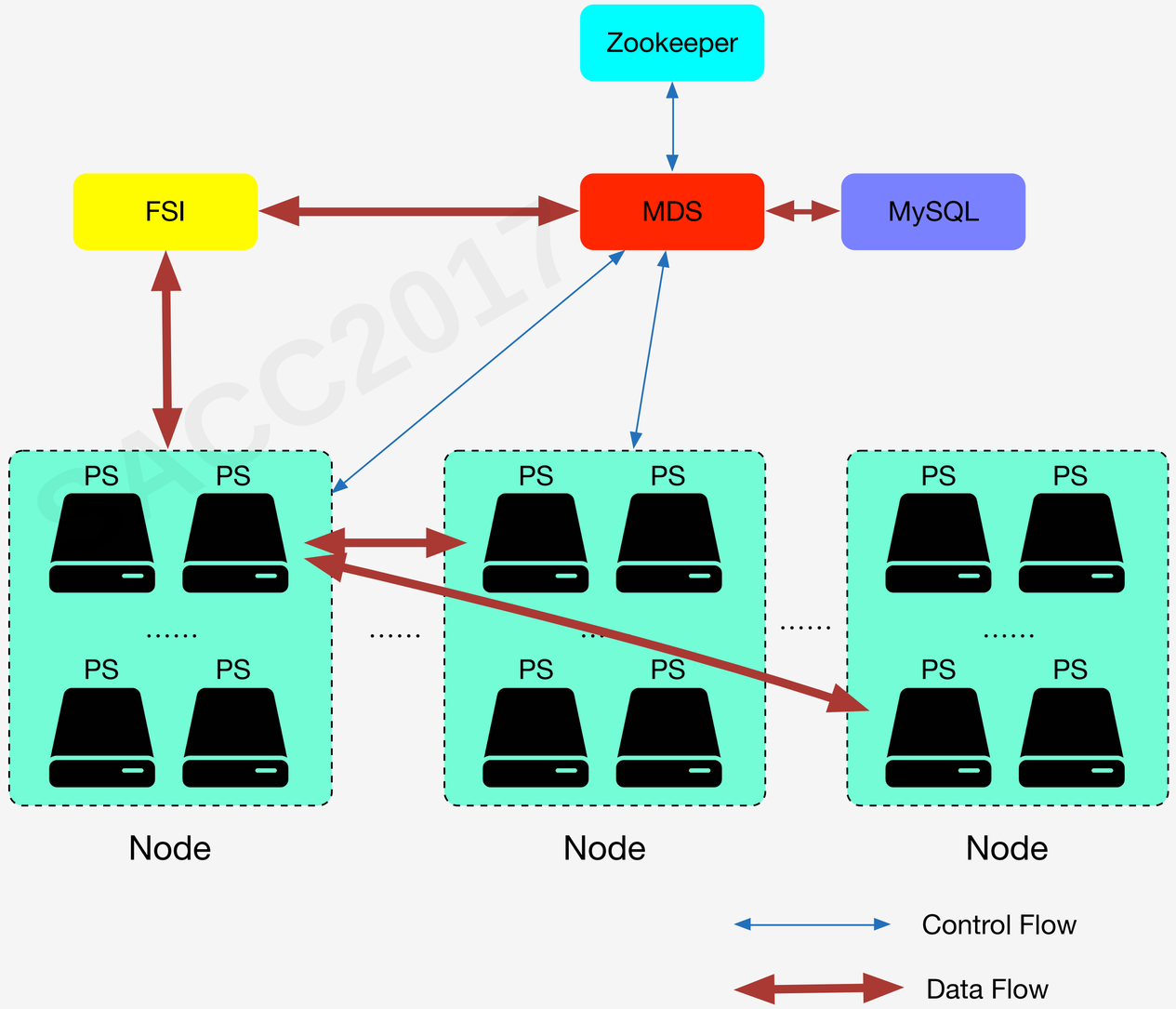
# Topology

- ✓ User
- ✓ Pool
- ✓ Zone
- ✓ server
- ✓ Disk

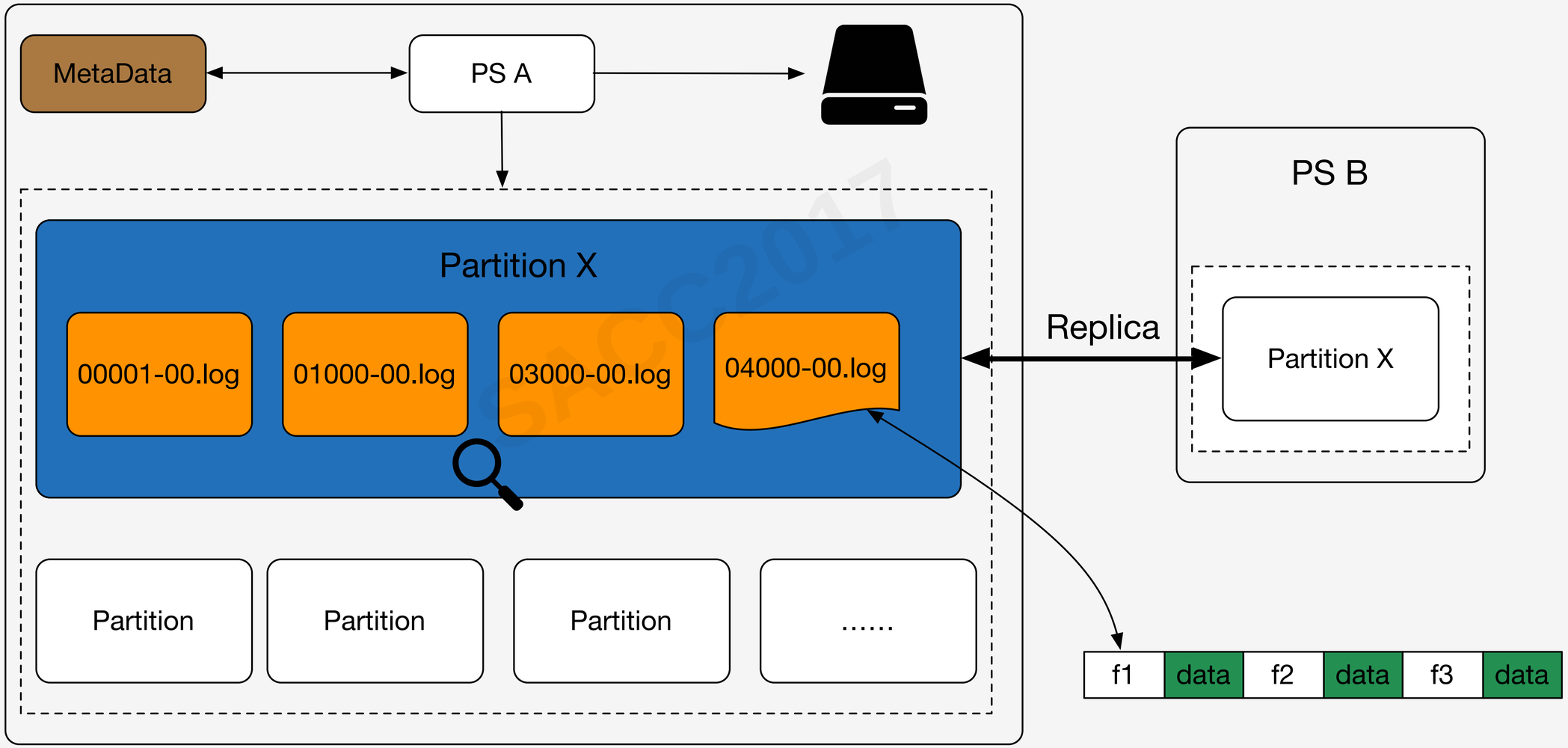



# Architecture

- ✓ PS：Partition Server
- ✓ MDS、MySQL
- ✓ ZooKeeper
- ✓ FSI



# PartitionServer



# MDS

- 数据定位：Topology 、（ PartitionID->“ps1-ps2-ps3” ）
- 数据分布、放置、均衡

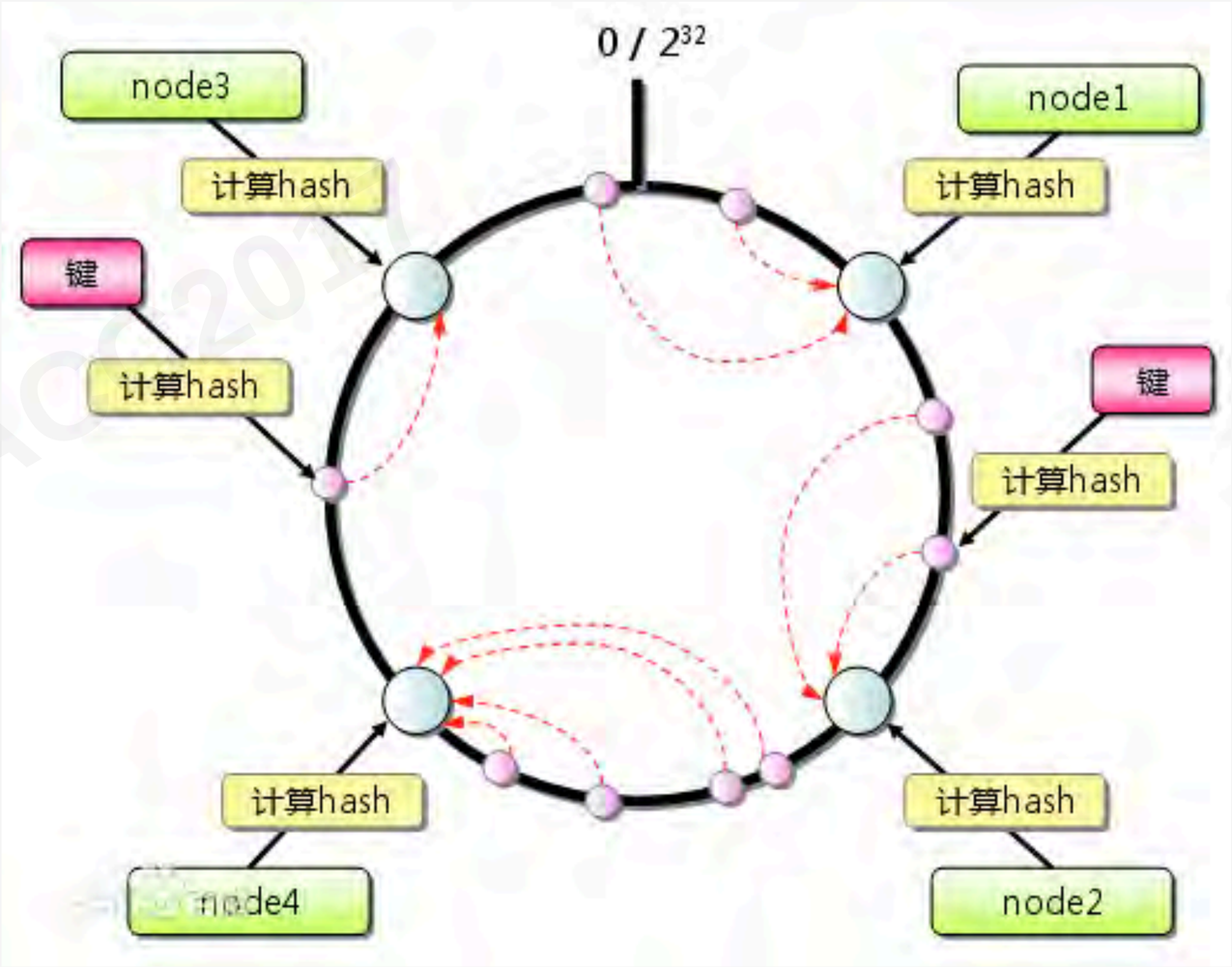
# VS

- 去中心化v元数据

  ✓consistent hash & Crush

  ✓元数据少

  ✓不够灵活：扩容、数据迁移

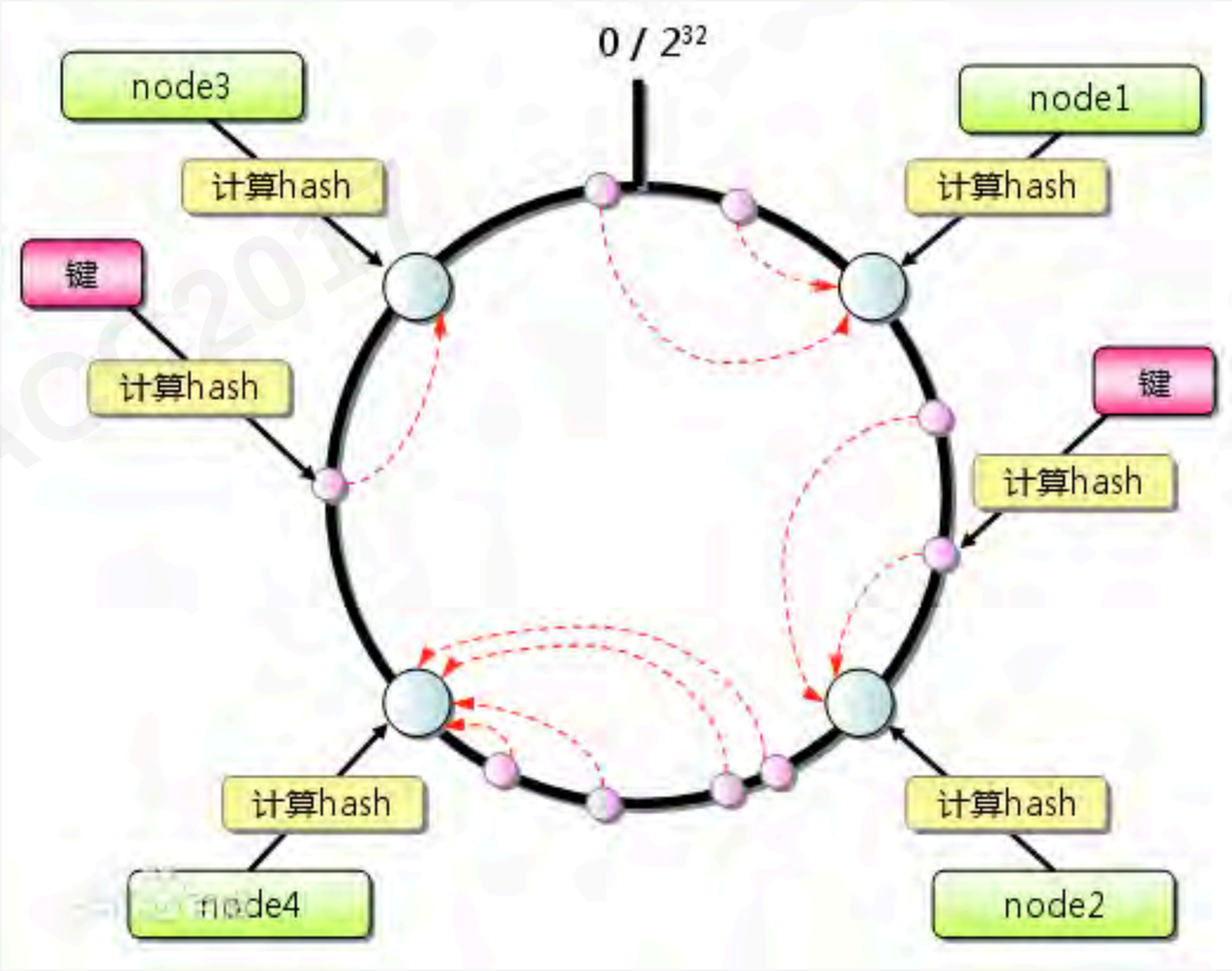


# Choose

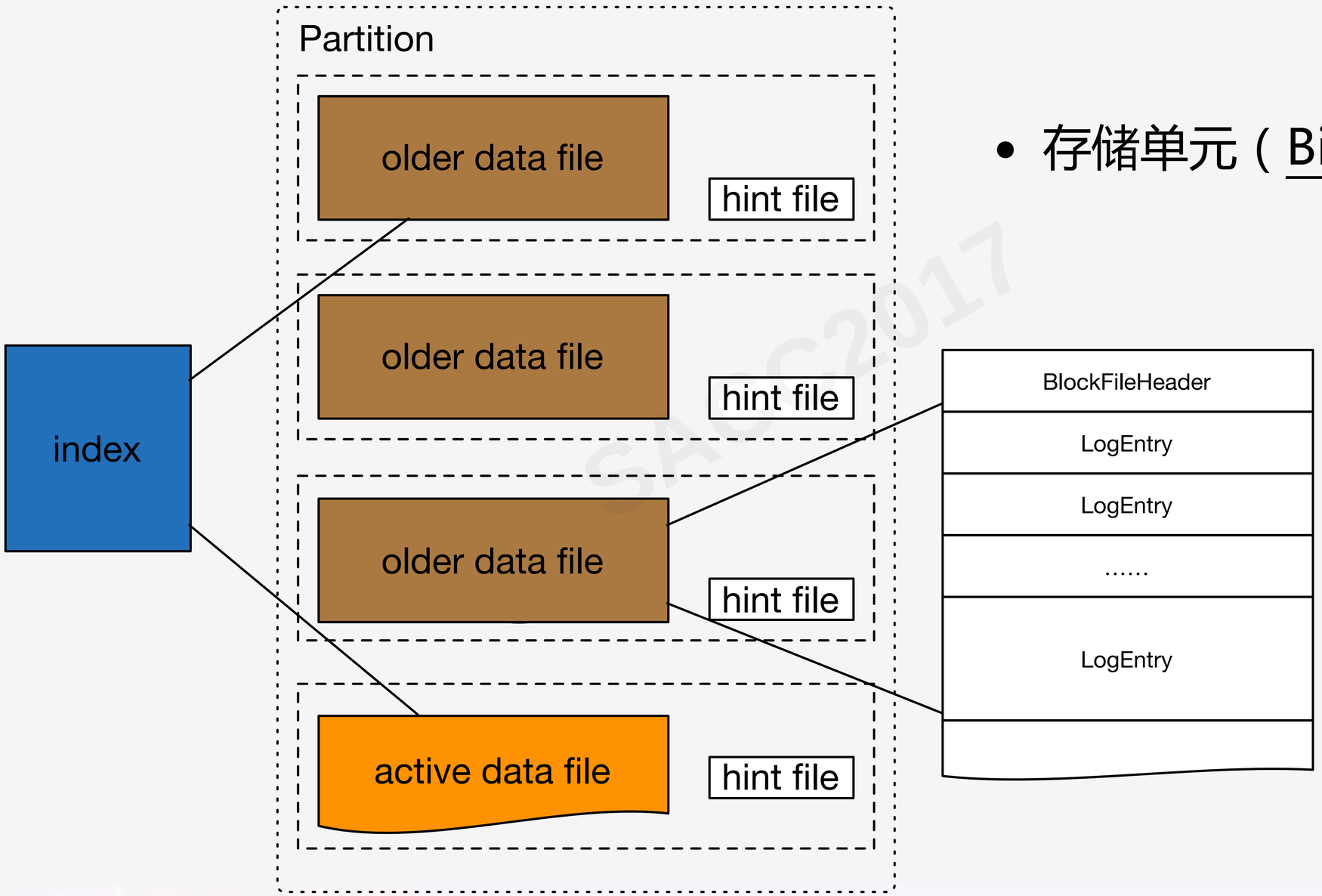
- Reality

  ✓元数据本来就少，100PB，几十M

  ✓按需扩容，不希望强制rebalance

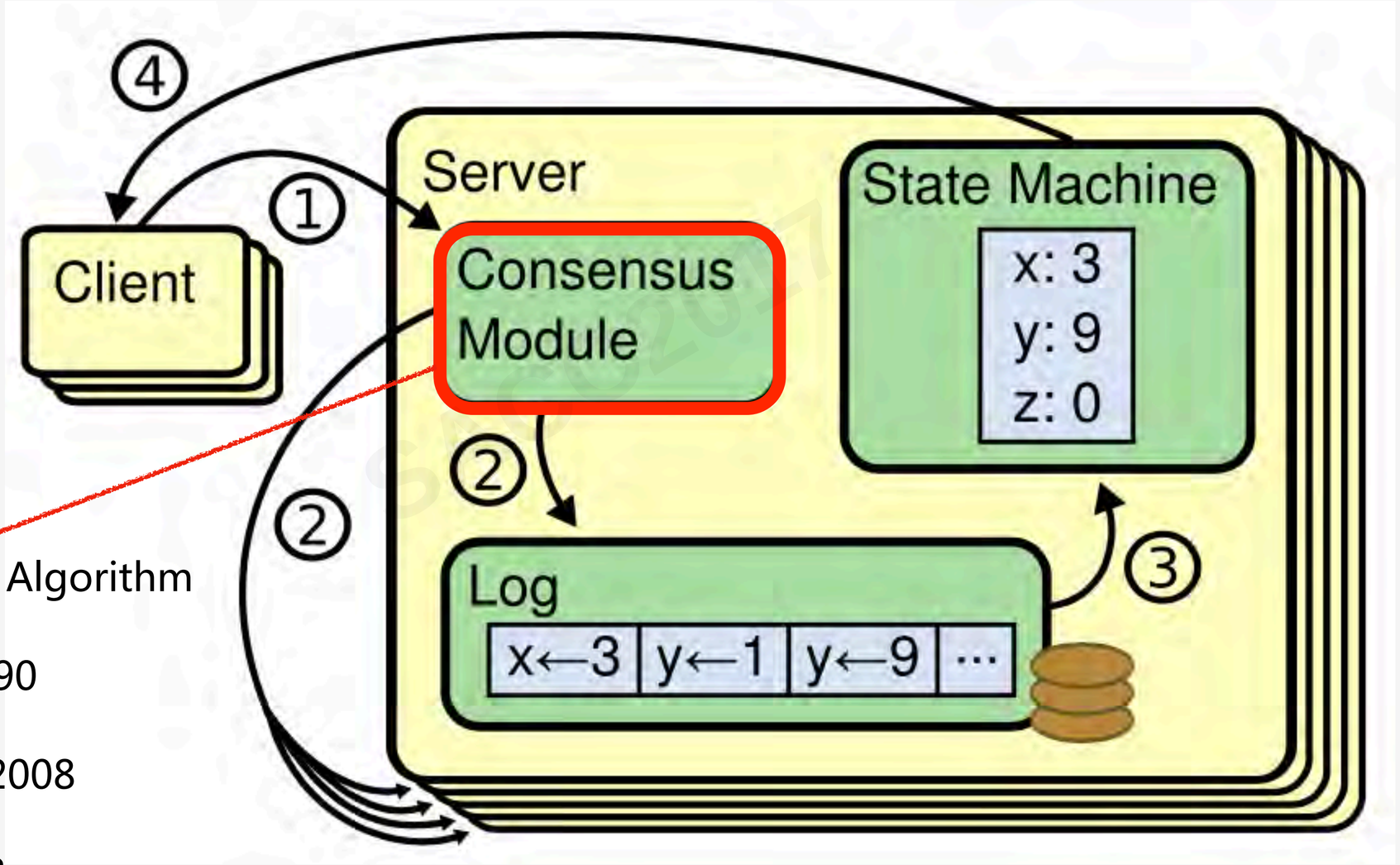


# NEFS




- 存储单元（BitCast存储模型）

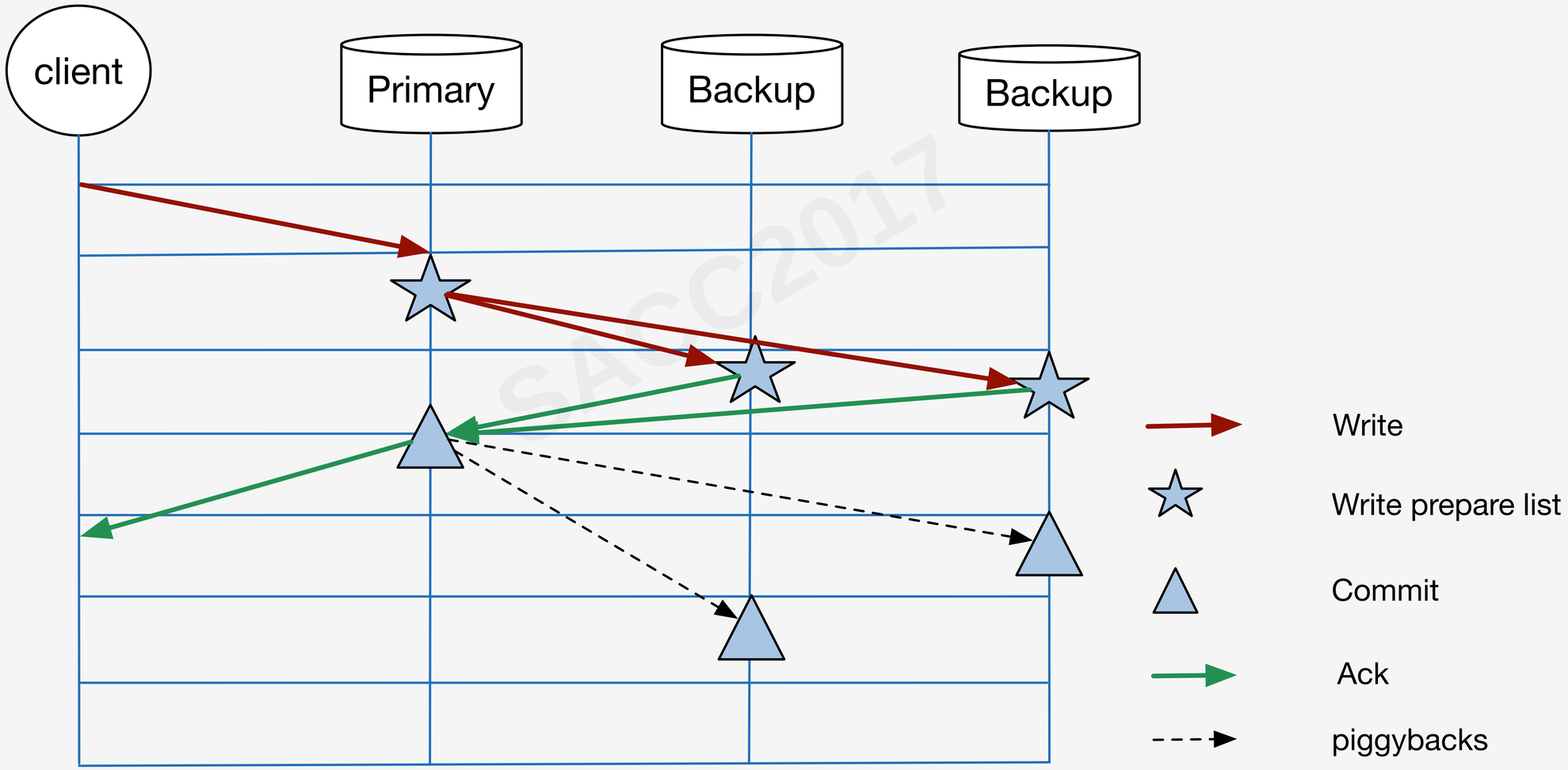
# 数据复制



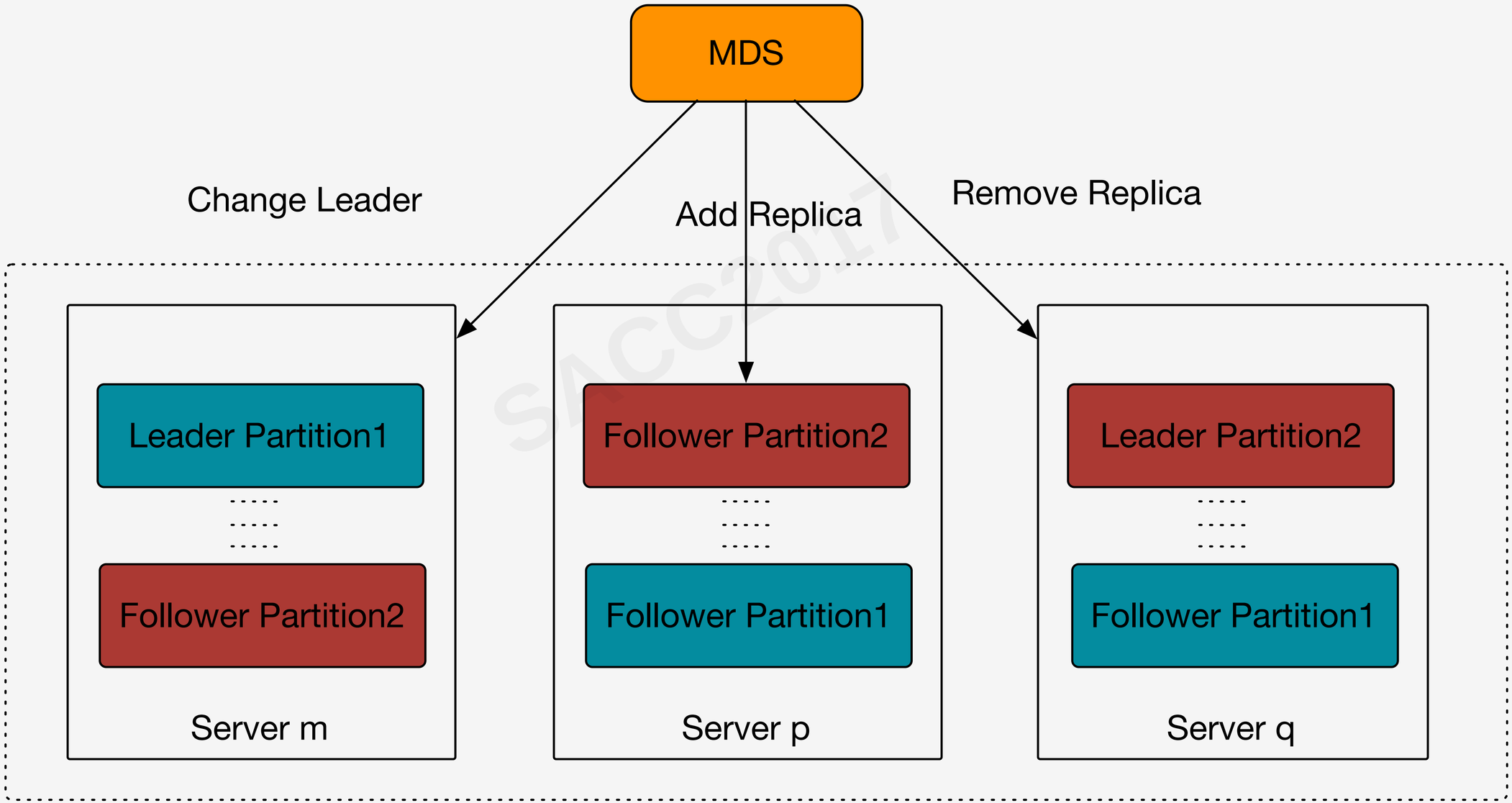

Replicated State Machine Architecture

- Consistency Algorithm
  - ✓ Paxos 1990
  - ✓ PacificA 2008
  - ✓ Raft 2013

# Basic PacificA



# MemberShip Change



# PacificA vs Raft

| VS | Basic | MemberShip | Performance | Avalibility | Durability |
|---|---|---|---|---|---|
| PacifcA | Write-ALL | 依赖外部 | Low | Low | High |
| Raft | 2/F +1 | 依赖自身 | High | High | Low |

# Choose

- Reality
  - Write Any Replica(Partition) Group
  - MDS in  System
  - Durability is important than Performance
  - Easy Implementation

# NEFS

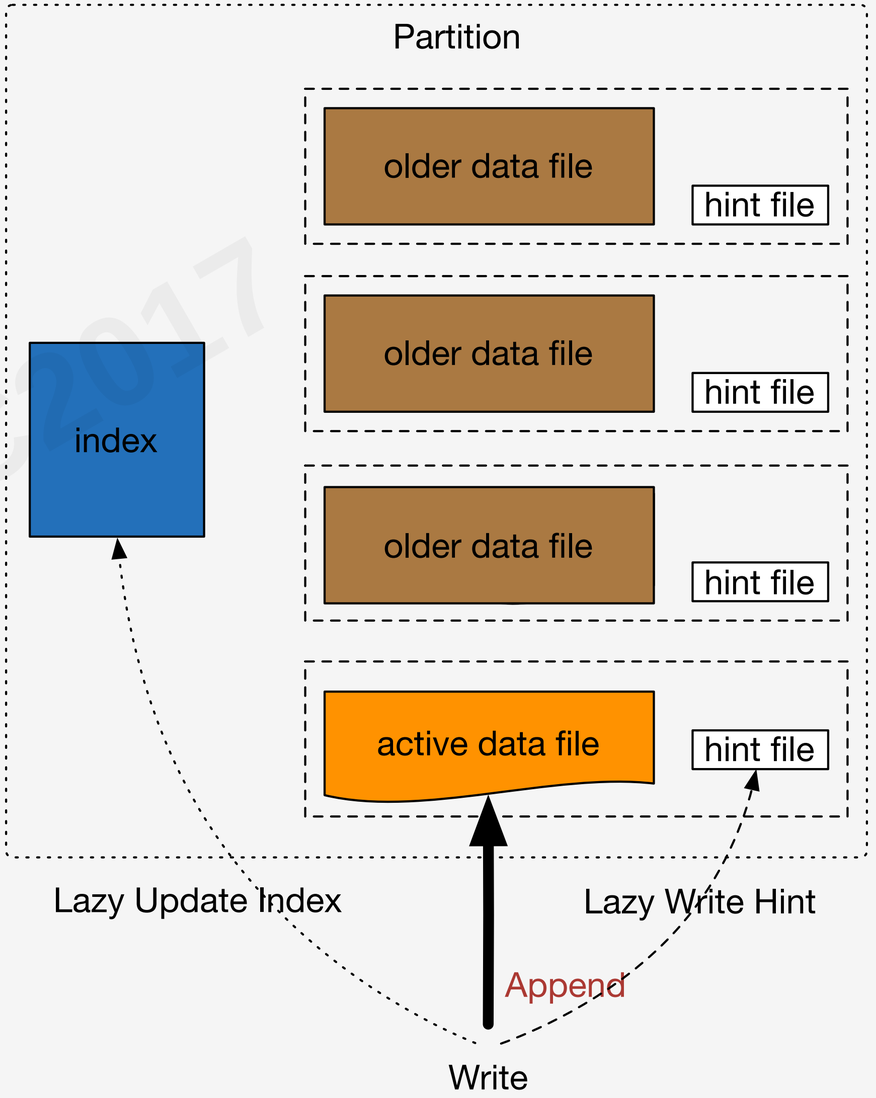
- Performance
- Durability
- Cost

# NEFS

- Performance
- Durability
- Cost

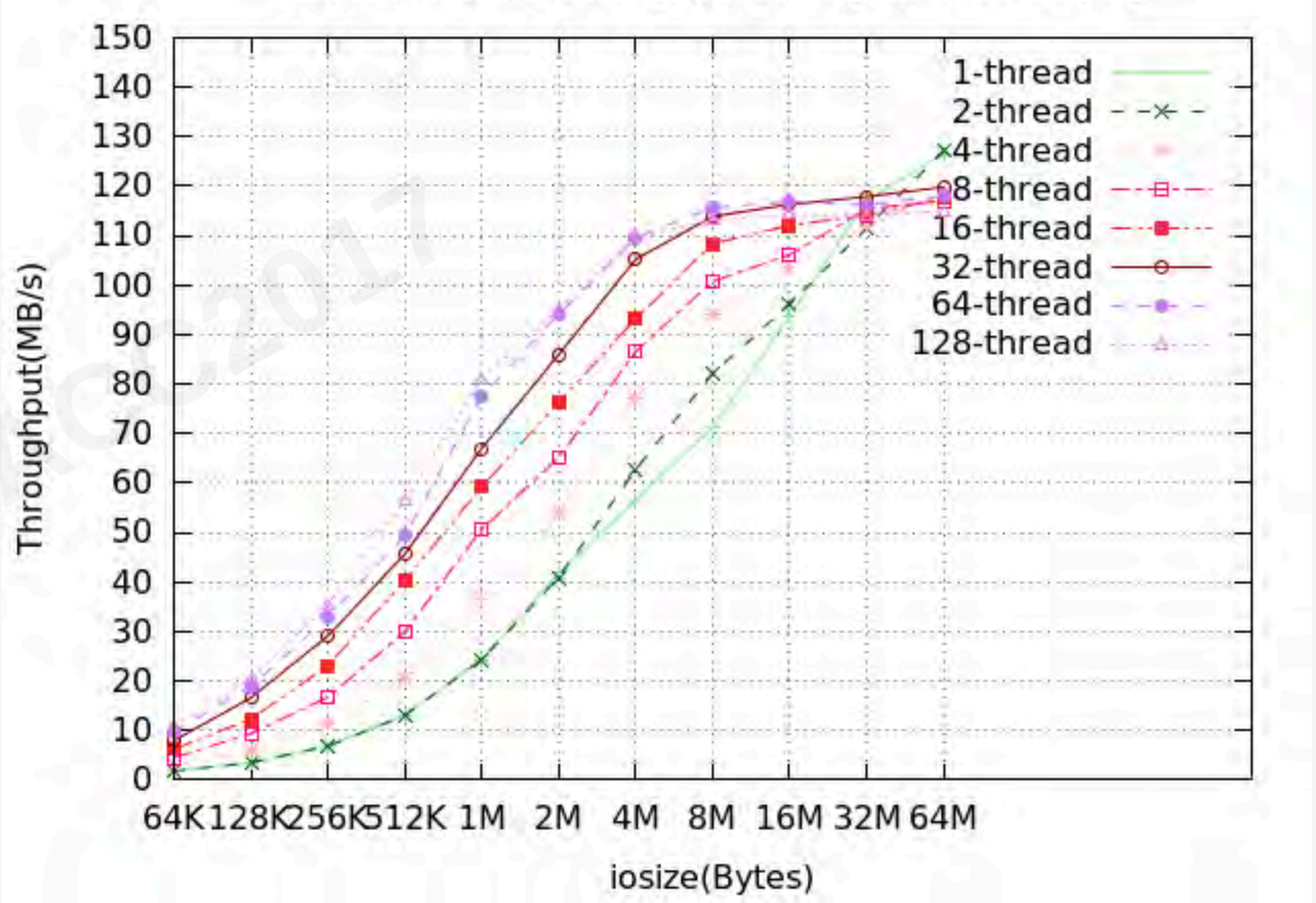
# Performance

- ✓ Just One IO Per Write
- ✓ Big File split Into  1MBs slice
- ✓ IO 优化
  - ✓ Limit Concurrent IO
  - ✓ GroupCommit
  - ✓ Delete Not Force Flush



# Performance

- ✓ Just One IO Per Write
- ✓ Big File split Into 1MBs slice
- ✓ IO 优化
  - ✓ Limit Concurrent IO
  - ✓ GroupCommit
  - ✓ Delete Not Force Flush




硬盘性能简测

# NEFS

- Performance
- Durability
- Cost

# NEFS

- Durability

| AWS 产品线 | SLA |
| --- | --- |
| S3 Standard | 11个9 |
| S3 Standard – IA | 11个9 |
| Glacier | 11个9 |

- 100亿文件一年只可能丢失1个文件

# Durability- 影响因素

- ✓ AFR：磁盘年故障率
- ✓ RepNum：存储复制因子
- ✓ T：坏盘恢复时间
- ✓ S：系统CopySet数量
- ✓ N：系统中磁盘数量

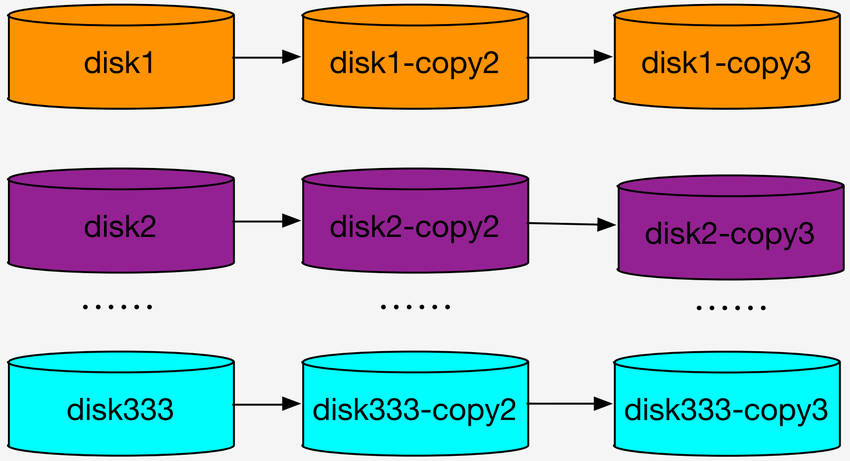
# Durability- 影响因素

- ✓AFR：磁盘年故障率
- ✓RepNum：存储复制因子
- ✓T：坏盘恢复时间
- ✓S：系统CopySet数量
- ✓N：系统中磁盘数量

“在包含999块磁盘的3备份存储系统中，同时坏三块盘情况下的数据丢失概率?”

**设计一**：把999块磁盘组成333个磁盘对。

333/C(999,3) =5.02*e-07



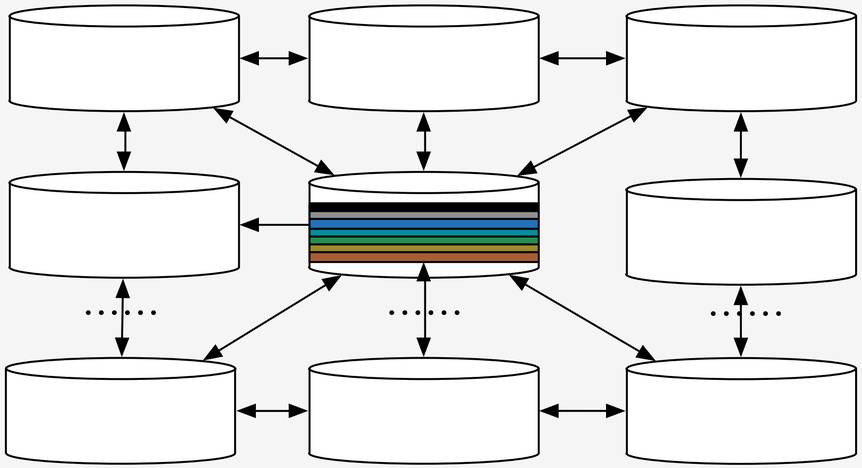
# Durability- 影响因素

- ✓AFR：磁盘年故障率
- ✓RepNum：存储复制因子
- ✓T：坏盘恢复时间
- ✓S：系统CopySet数量
- ✓N：系统中磁盘数量

“在包含999块磁盘的3备份存储系统中，同时坏三块盘情况下的数据丢失概率?”

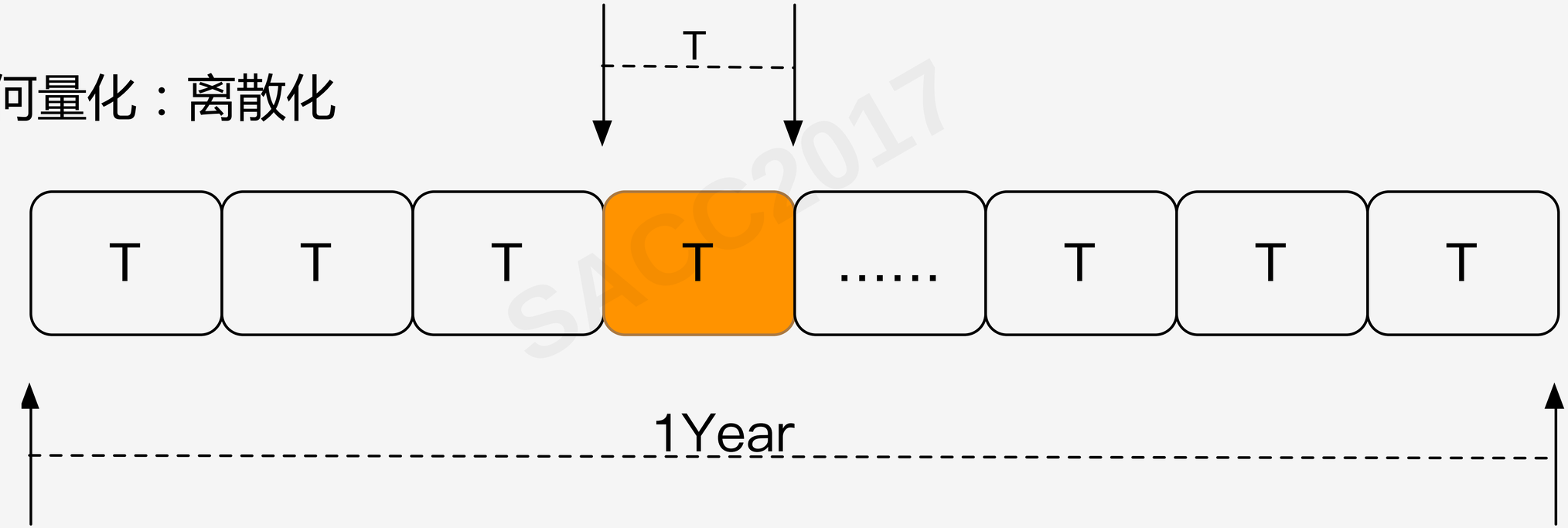
**设计二**：数据随机打散到999盘中

C(999,3)/C(999,3)=1



# 如何度量

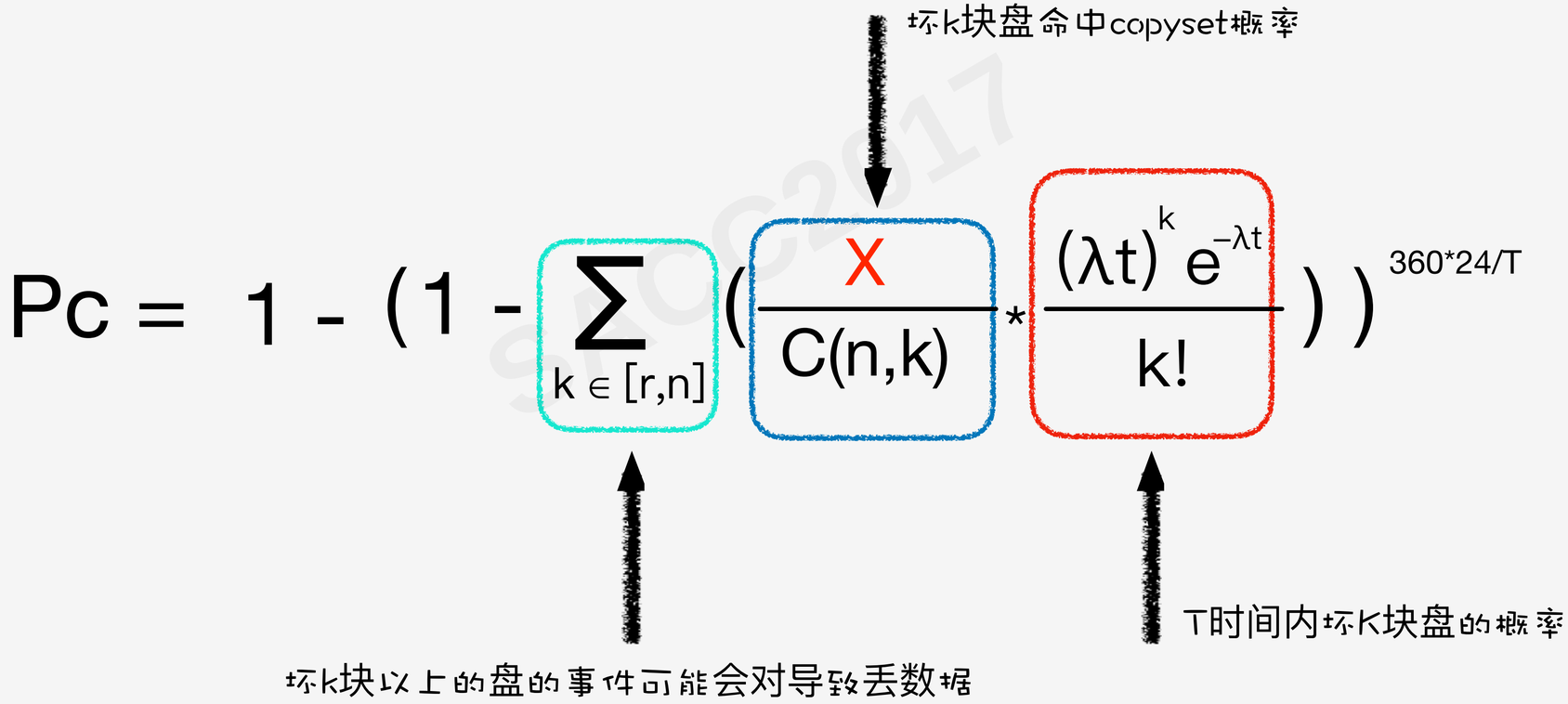
- 如何量化：离散化






$$Pc = 1 - (1-Pb(T))^{**}(365*24/T)$$

# 如何度量






分布式存储可靠性

# NEFS

- Durability设计考量
  - ✓副本数
  - ✓恢复时间
  - ✓CopySet数量

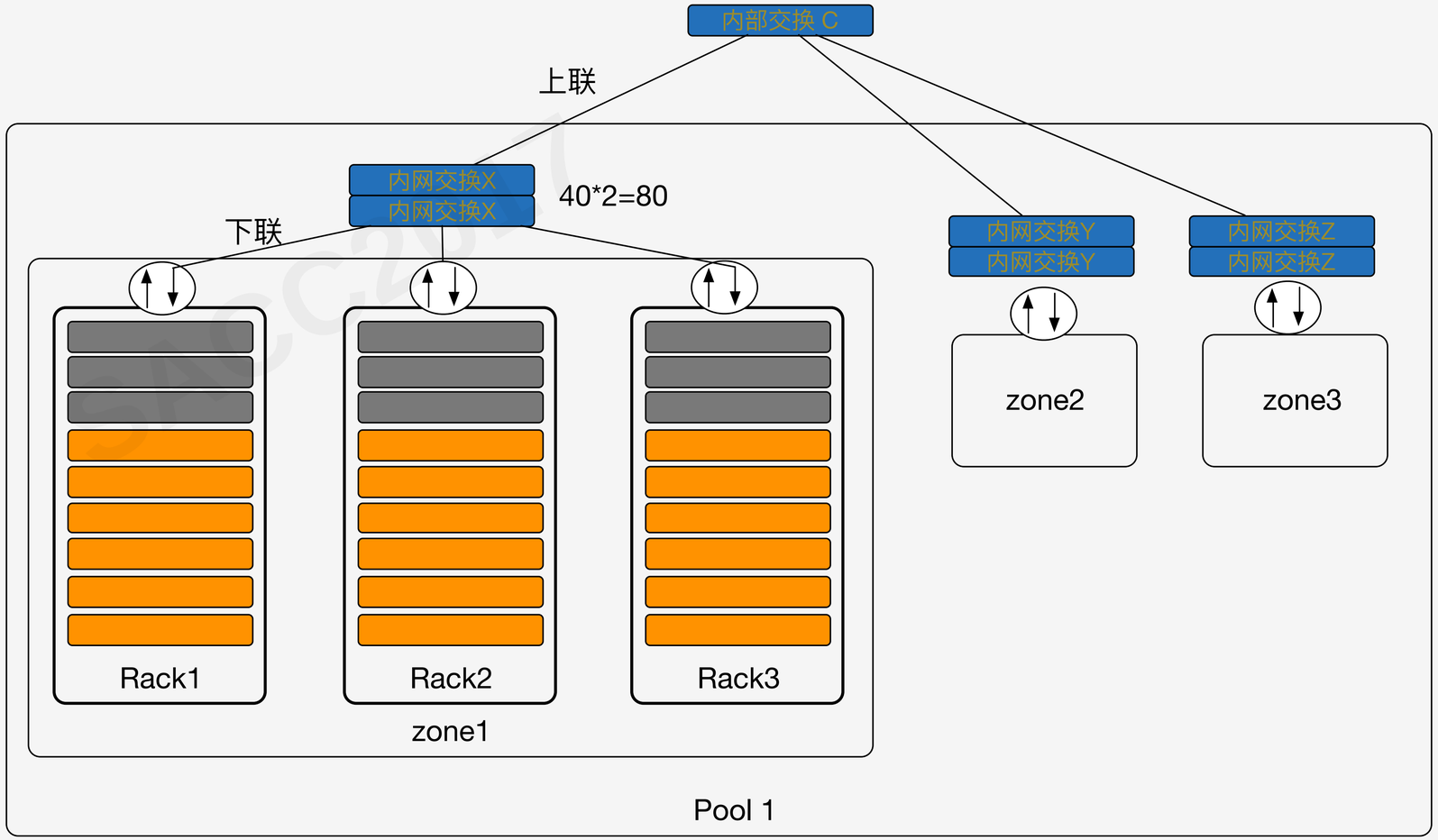
# NEFS

- 恢复时间

  ✓布局

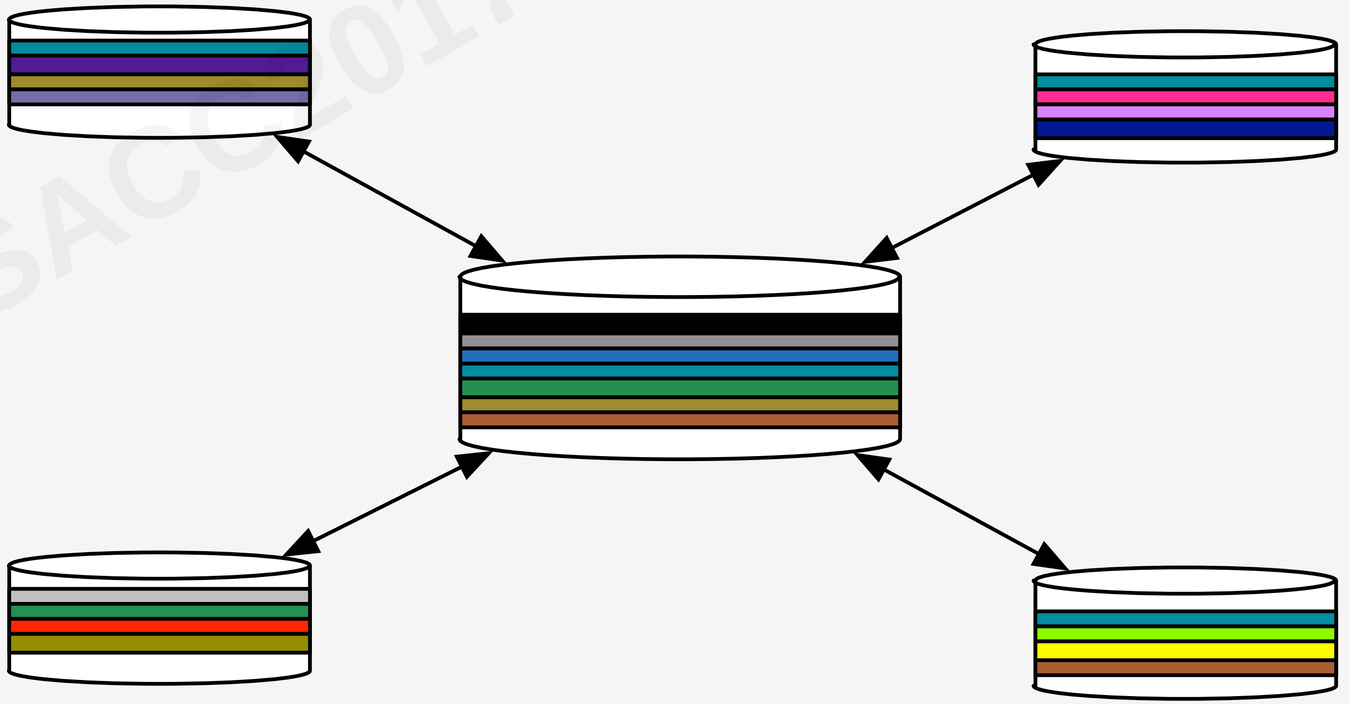
  ✓复制单元放置&大小

  ✓网络IO限速



# NEFS

- 恢复时间
  - ✓布局
  - ✓复制单元放置&大小
  - ✓网络IO限速

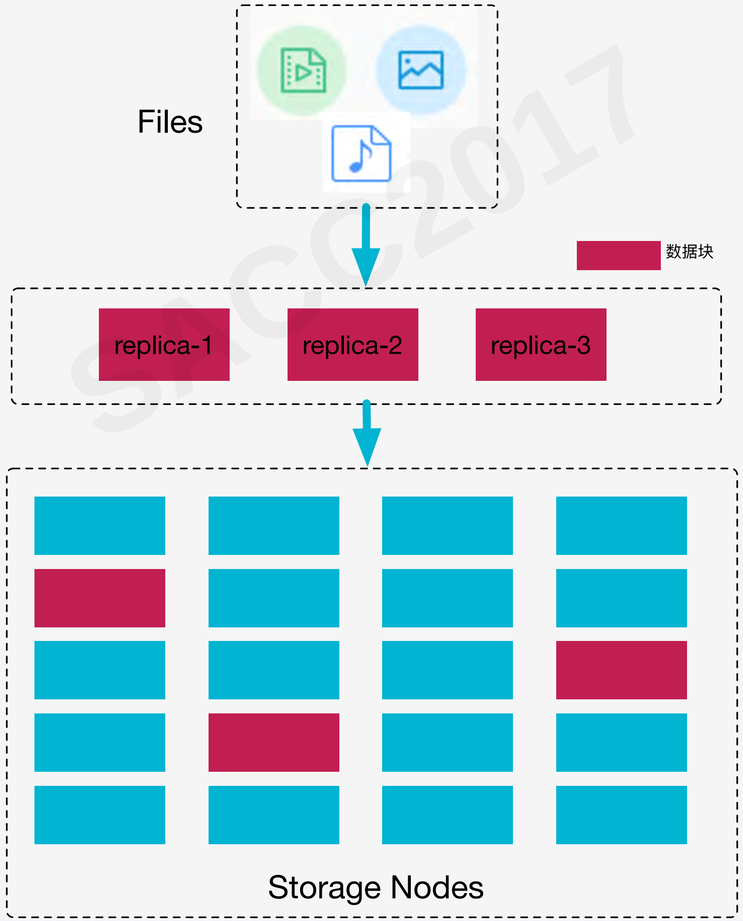


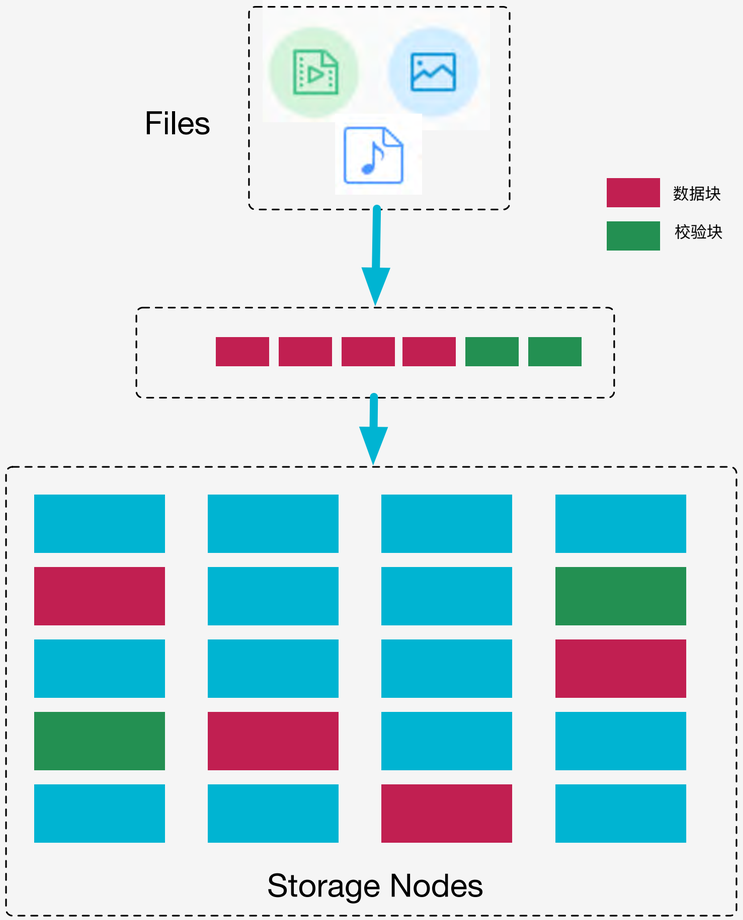
# NEFS

- Performance
- Durability
- Cost

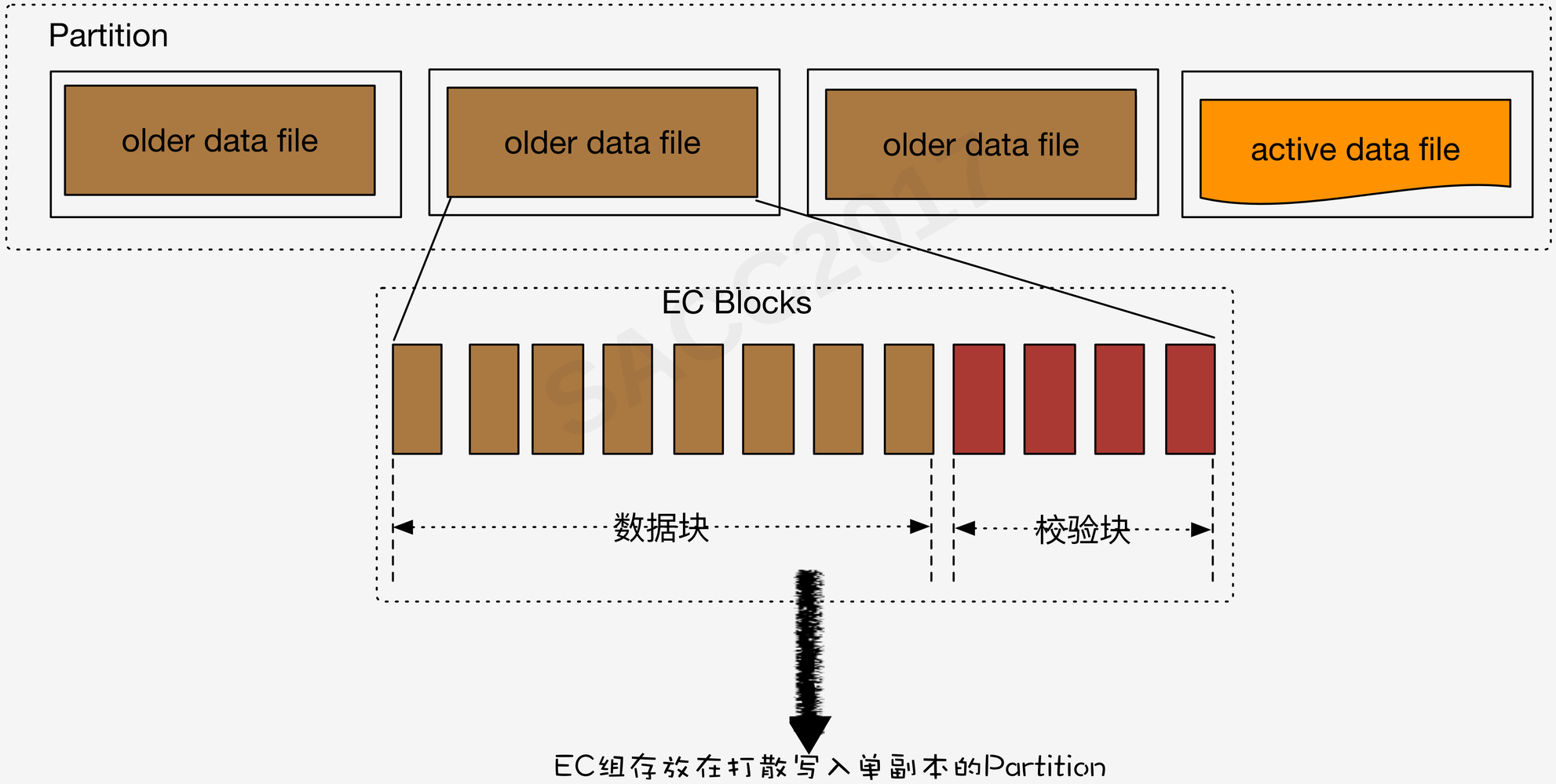
# NEFS

- Cost
  - 提高复制因子
    - 副本技术
    - EC



# NEFS


Partition
older data file
older data file
older data file
active data file
EC Blocks
数据块
校验块
EC组存放在打散写入单副本的Partition

# 总结

- A scalable high-available log-based Distributed Key-Value  Blob Storage system.

  ✓Key-Value ：Put、Get、Delete

  ✓Storage Engine： Log-Based(BitCase)Storage Engine

  ✓Strong Consistent(PacificA )

  ✓Durability: 3 Copy Replica & Erase Code

  ✓It is Simple

# Future Works

- Load Balance
- EC Enhance
- Performance
- Metric、Ops
- Remove zookeeper & Mysql
- ……

# Hiring


THANKS